A THEORY OF INFORMATION PROCESSING FOR MACHINE VISUAL PERCEPTION: INSPIRATION FROM PSYCHOLOGY, FORMAL ANALYSIS AND APPLICATIONS.

A Thesis submitted for the degree of:
Doctor of Philosophy

Escuela Politécnica Superior
Departamento de Ingeniería Informática

UNIVERSIDAD AUTÓNOMA DE MADRID

Author: Eduardo Cermeño Mediavilla
Madrid, June 2017

Adviser: Juan A. Sigüenza Pizarro

To my family.
Nulla dies sine linea

Acknowledgements

I would like to thank all the people who made this thesis possible. I thank my academic adviser, Professor Juan A. Sigüenza, for giving me the opportunity to find my way towards a deeper understanding of artificial intelligence. His advice has been the reference that has allowed both to explore different fields of knowledge related to computer vision and to complete this dissertation. I would also like to thank Professor Benjamín Sierra for inviting me to discover why psychology matters for computer vision.

I am grateful to Helena, Nadia and Professor Norberto Cerezal for their support in improving the dissertation.

This work is the result of answering questions that arise day by day in the company that I co-founded more than ten years ago. This thesis is somehow a formalization of the effort that we make every day to create machines that see. I wish to thank my colleagues for their work which has been a great inspiration.

I want to acknowledge the work of so many good teachers, from whom I have learned so much. Additionally, I would like to remember all the people that I have met during this time, and that in some way have contributed to this thesis.

Finally, I thank my parents Fortunato and Benita for all the sacrifices they have made to support my studies. This work is just the continuation of what they started.

Abstract

Computer vision is the branch of artificial intelligence concerned with enabling computers to understand images and videos. The fields of application are diverse and solutions have been implemented to automatize different problems. Despite some impressive achievements, computer vision applications undergo important limitations if compared with human vision. Our objective is to understand the reasons why computer vision results are often behind those of human vision.

We need to understand why we see what we see and how reliable is it. The results of visual perception are a selection of statistical reflections of visual history and not a veridical representation of the physical world. Our false sensation of certainty is a consequence of a stable world, in which things change but maintain a certain degree of invariance. Our visual system is able to detect these invariant properties and relate them to represent the physical world.

Computer vision state-of-the-art methods classify sets of features to recognize objects. Our thesis statement is that pattern classification cannot explain by itself the variety of results from human vision. What is perceived is not only a function of the elements on the image but also includes the knowledge of the perceiver and what has been perceived before. We propose that perception is a process of information gathering, which could be approached as a search problem, and addressed by an intelligent agent.

We suggest that what is perceived are categories, which are sets of objects, each of them defined by a set of constraints relating properties. Thus, any relation of properties might be considered as the definition of a category, allowing to categorize anything with a form. Different kinds of computer vision problems can be approached by categorizing the whole without categorizing the parts. Direct categorization of the whole is in many cases more reliable and efficient

than an indirect one based on the comprehension of the categorization of the parts. Machine visual systems adapt through a continuous process of integrating the collected information.

Resumen

La visión artificial es la rama de inteligencia artificial que se ocupa de permitir que los ordenadores puedan comprender el contenido de imágenes y videos. Los campos de aplicación son diversos y ya se han implementado soluciones para automatizar diferentes tipos de problemas. A pesar de algunos logros impresionantes, las aplicaciones de visión artificial sufren de importantes limitaciones en comparación con la visión humana. Nuestro objetivo es comprender las razones por las que los resultados de aplicaciones de visión artificial suelen estar por detrás de los obtenidos por la visión humana.

Necesitamos comprender por qué vemos lo que vemos y su fiabilidad. Los resultados de la percepción visual son una selección de reflexiones estadísticas de la historia visual y no una representación verídica del mundo físico. Nuestra falsa sensación de certeza es la consecuencia de un mundo estable, en el cual las cosas cambian, pero también mantienen un cierto grado de invarianza. Nuestro sistema visual es capaz de detectar estas propiedades invariantes y relacionarlas para representar el mundo físico.

Los métodos del estado del arte en visión artificial clasifican conjuntos de características para reconocer objetos. Nuestra tesis afirma que la clasificación de patrones no puede explicar por sí sola la variedad de resultados que ofrece la visión humana. Lo que se percibe no depende únicamente de los elementos de la imagen sino que también depende del conocimiento del perceptor y de lo que ha sido percibido con anterioridad. Proponemos que la percepción es un proceso de recogida de información que puede enfocarse como un problema de búsqueda y abordarse mediante agentes inteligentes.

Sugerimos que lo que se percibe son categorías, las cuales son conjuntos de características, cada una definida por un conjunto de condiciones que relacionan

propiedades. De esta forma, cualquier relación de propiedades puede considerarse como la definición de una categoría, permitiendo categorizar cualquier cosa con forma. Diferentes tipos de problemas de visión artificial se pueden abordar mediante la categorización del todo sin categorizar las partes. La categorización directa del todo es en muchas ocasiones más fiable y eficiente que la categorización indirecta a través de la comprensión de la categorización de las partes. Los sistemas de percepción visual automática deben poder adaptarse a través de un proceso continuo de integración de la información recopilada.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The visual world is likely the main source of information for humans. We use it to move around, to find food or friends, to avoid dangers or just to learn new things. Representations of the visual world, paintings, pictures and more recently video are an important part of our lives. People enjoy painting, collecting art, visiting galleries, taking pictures or going to the cinema. We have successfully developed tools to introduce these representations into computers, which can now easily store, display or transmit them. Computers are in fact one of the main tools to create or edit images, what is known as computer graphics.

Many movies include computer graphics achieving impressive results, which would be difficult to reach by humans without the support of machines. On the other hand something that healthy humans do effortlessly, understanding what images represent, is still a big challenge for machines. Understanding the content of images is fundamental to implement image retrieval systems, automatize surveillance tasks or to develop intelligent agents like self-driving cars. The research field concerned with image understanding is referred to by different names, computer vision, machine vision or visual perception. Some authors differentiate between computer vision and machine vision (Davies [2008] p.13) but the difference is questionable. For us the only difference will be image acquisition, computer vision only deals with digital images while machine vision includes the techniques for digitization. Unless stated otherwise, in this dissertation we will use them indistinctly. In fact in many cases we would rather use machine visual perception because it includes the word perception instead of vision. The mean-

ing is the same, but vision is usually associated to sensors, and we will explore the differences between sensors and perceptual systems.

Machine perception is usually related to artificial intelligence (AI), and this dissertation seeks to better understand the relationship between them. At first sight the concept of AI might seem easy to understand, AI is just about making machines emulate human intelligence. The problem is that even though human intelligence is something familiar for most people, the answer to the simplest question *What is intelligence?* is not so simple. Instead of considering the nature of intelligence, very often we just evaluate the intelligence of a machine by comparing it with the equivalent human actions. A machine that plays chess is likely considered an intelligent machine, whereas one that cuts plastic pieces is not. In fact the evaluation of a machine's intelligence can change over time, for example optical character readers used to be considered as AI programs in their beginning, but when results reached sufficient reliability they lost their "intelligent status" (Schank [1991]).

The example of optical character readers can be extrapolated to many applications of computer vision, whether it is an intelligent machine or not is usually determined by how impressive the activity undertaken by the machine is. In this dissertation we suggest that any machine able to perceive could be considered an intelligent machine when perception is understood as a process of information gathering. This PhD Thesis explores the fundamentals of vision to understand how visual perception systems can be built to emulate or improve the results given by human vision.

1.1 Computer vision

1.1.1 Applications

Computer vision is attracting much interest. Today it is likely the most active research field within artificial intelligence. Maybe the main trigger for such a hectic activity is the evolution of hardware, which enables working with images in a way unseen before. Affordable computers can store massive amounts of images and videos. The resolution of digital cameras is measured in millions of

pixels. Even low-power CPUs are now able to reproduce high quality video on mobile devices. Millions of images are taken and uploaded everyday. Video is everywhere. Such an amount of visual information cannot remain inaccessible to computers, it needs to be exploited.

Computer vision has applications in different fields like document analysis (Cermeño *et al.* [2014a]; He & Schomaker [2015]; LeCun *et al.* [1989]), video surveillance (Buch *et al.* [2011]; Cermeño *et al.* [2017b]; Hu *et al.* [2004]), food quality evaluation (Sun [2016]), sports analysis (Moeslund *et al.* [2015]) or affective computing (Perez *et al.* [2014]; Picard [2000]). These applications are already part of our daily life with products like business card readers [1], game consoles [2] or autonomous driving cars [3].

(a) Xbox Kinect (b) Business card reader

Figure 1.1: Computer vision applications

Such products have computer vision systems that basically fulfill one of the following four functions:

- Object [4] detection

[1] www.abbyy.com
[2] www.xbox.com/es-ES/xbox-one/accessories/kinect
[3] www.tesla.com
[4] The word object should be understood in a broad sense, an object may be an animal, a person or a face

- Object tracking
- Scene parsing
- Scene classification

Object detection methods seek to find known objects within an image, while in a video sequence tracking methods relate objects from a frame to the objects from previous frames. Scene parsing methods are closely related to object detection, however the latter only search for a set of known objects in the image, while the former try to divide the image into regions associated with semantic categories such as person, car, sky, grass etc. Object detection reports the position of the object when it is found, while scene parsing reports a description of the scene. Finally scene classification methods assign a label to an image or video sequence. *But how could a machine fulfill such functions ?*

1.1.2 Techniques

Answers to the previous question are usually found in *Digital Image Processing* and *Pattern Recognition* literature. The distance between image processing and computer vision is not clear. Some of the most cited books in the field are somehow based or consider a useful paradigm that divides computerized processes into three types: low-level (early), mid-level (intermediate) and high-level (Davies [2008]; Forsyth & Ponce [2003]; Gonzalez & Woods [2008]). Low-level vision deals with image transformations, such as noise removal filters or morphological operations like erode or dilation, and feature extraction, such as edge detection or texture analysis. Mid-level vision is concerned with extracting information about the images, such as shapes and motion. High-level vision involves pattern recognition, establishing a relationship between image features and object or scene features.

The nomenclature of the paradigm suggests the idea of sequential processing: first low-level, then mid-level and finally high-level processing. Figure 1.2 reproduces a diagram from Szeliski [2010] describing the relationship between different techniques in computer vision. It also suggests a sequential processing, segmentation and feature detection connected from one side to image processing and to

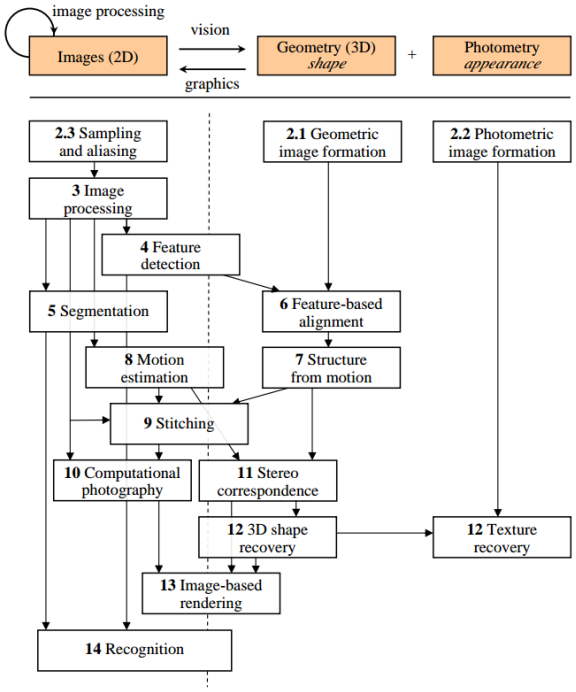


Figure 1.2: Relationship between images, geometry, and photometry, as well as taxonomy of the topics covered in Szeliski [2010]

recognition on the other side, in what could be seen as three levels. However the author warns that "this taxonomy should be taken with a large grain of salt, as the processing and dependencies in this diagram are not strictly sequential" (p.19).

1.2 Motivation of the Thesis

Eduardo Cermeño has worked in a company specialized in computer vision applications since 2004. Everyday, people and companies show their interest in automatizing a wide range of tasks such as those involving vision, from quality verification to behavior analysis. For instance companies wish to know how many people go into their shops, what are the most visited areas, how long do customers have to wait in a queue before paying, even their mood when leaving the shop. Human observers could collect information to answer all these questions,

but could a machine do it? In this dissertation we deal with the fundamental questions that need to be solved to understand how machines could emulate or improve the results of human visual perception: *what has to be perceived?*, *how does a machine perceive it?* and *how do we build such a machine?*

We bound the first question by considering it equivalent to *could a machine perceive everything that is perceived by a human?* The answer to this question requires knowledge about *what humans are able to see.* Answers to the second one should propose a strategy for perceiving whatever has been answered in the first question, and explain why such a strategy is appropriated.

Computer vision literature presents many techniques but does not explain their role in the process of visual perception. For example, we know that segmentation divides images into parts, but *why should we need to divide an image into parts ?* Some authors consider in their books object recognition to be a high-level process (Davies [2008]; Forsyth & Ponce [2003]), whereas others (Gonzalez & Woods [2008]) consider it an intermediate process, but if we are interested in scene classification *why should we perform object recognition ?* However paradoxical it may seem, we have not found an explicit computational theory for machine visual perception, that explains what is computed and why. The same problem was tackled by Marr [1982] for human vision. The way Marr [1982] approaches vision has been very inspiring. The same questions and methodology used for understanding human vision can be used to better understand computer vision.

Marr [1982] suggests that neurophysiological findings are not enough to understand human vision, the present Dissertation questions whether research in new features or classifiers is sufficient to understand how a perceptual systems comparable to human vision could be designed. At the beginning of the century Viola & Jones [2001] and Lowe [2004] presented two promising methods for extracting features for object recognition. In 2012, after the publication of the dataset Imagenet (Deng *et al.* [2009]), a different approach was presented by Krizhevsky *et al.* [2012], starting a new wave of methods based on convolutional neural networks, that have surpassed previous state-of-the-art methods for object recognition (Girshick *et al.* [2014]; He *et al.* [2016]; Sun & Ponce [2016]) and scene parsing (Grangier *et al.* [2009]; Karpathy & Fei-Fei [2015]).

Very often machine visual perception is seen as a pattern recognition problem.

If this were the case we would not be far away from the solution. Simonyan & Zisserman [2015] achieves a top-5 error rate of 6.8 % in the Imagenet Large Scale Visual Recognition Challenge - ILSVRC- (Russakovsky *et al.* [2015]). This means that a proportion of 93.2 % of the images had their ground-truth label among a set of 5 predictions given by their algorithm. The ILSVC test set has 100.000 images with 1000 categories covering plants, geological formations, natural objects, sports, artifacts, fungus, people, animals, food etc.

However the reality is that we are not as close to finding a solution as these results might let think. Real world applications very often go beyond object recognition. People are able to perceive birds in the sky or in videos, even if they are a few pixels in size. People are able to distinguish between a moving tree and a human intruder, even under a costume. They are able to recognize the effects of an illumination change, even if they have never seen a change like that before. The present PhD Thesis is motivated by the experience acquired in a company that develops computer vision applications and the will to explore fundamental questions for which no answer has yet been found.

Russell & Norvig [2014] states that some influential founders of AI (Beal & Winston [2009]; McCarthy [2007]; Nilsson [2005] "have expressed discontent with the progress of AI". They think that research in AI should focus less on "ever improved versions of applications that are good for a specific task" and "return to its roots": "machines that think, that learn and that create" (p.27). Our research is about *machines that see*, about understanding what is required to make machine visual perception comparable to human vision. We are not searching new methods for solving a particular task involving vision, nor a general method to implement visual perception, we are searching for a theory that let us explain why the results of such or such computer vision system is not able to achieve the same results as human vision.

1.3 The Thesis

The Thesis developed in this dissertation proposes a theoretical general framework for explaining which computations are required by machine visual perception to achieve the results of human vision. It could be stated as follows:

Machine visual perception is an iterative heuristic process by which information related to an image is collected. The process combines top-down and bottom-up approaches to transform a set of pixels into a hierarchy of categories. Low level features are computed to recognize what has been seen before, while high level features are computed to comprehend what is been seen. A visual perception system is an intelligent agent whose program has three basic operators: segmentation, recognition and reasoning, and whose objective is to determine whether an image or its parts satisfy the conditions of a set of target categories.

1.4 Outline of the dissertation

In order to understand how a machine could achieve the results of human vision, the first step should be understanding the nature of those results. One of the objectives of this dissertation is analyzing the main theories about human visual perception. The neurophysiological basis of human vision is often present in the introduction of books on computer vision, and has inspired several methods applied in this area, such as Artificial Neural Networks (ANNs). The study of neurons involved in visual perception shows how human vision is biologically implemented but may not be enough to understand what is perceived or why this implementation is appropriate. Neurophysiology is closely related to psychology, the branch of science dealing both with mind and perception. We have reviewed relevant works from the field of psychology in search of answers to questions like *"why do things look as they do?"* (Koffka [1935]) or *"why do we see what we do?"* (Purves & Lotto [2003]). Psychology analyzes the processes of the mind behind vision and explain the logic of using such processes for vision, not only how they could be implemented. Marr [1982] outlines these different levels of explanation in table 1.1.

The second objective is to present a theoretical framework for explaining which computations are required to achieve the results of human vision. Visual perception is approached as an information processing activity, of which we analyze both the input and output. Based on this analysis we propose an algorithm with the actions required to perform the transformation of the input into the output. The theoretical framework deals with the levels of explanation called "Computational

Computational theory	Representation and algorithm	Hardware implementation
What is the goal of the computation, why is it appropriate, and what is the logic of the strategy by which it can be carried out?	How can this computational theory be implemented? In particular, what is the representation for the input and output, and what is the algorithm for the transformation?	How can the representation and algorithms be realized physically?

Table 1.1: The three levels at which any machine carrying out an information processing task must be understood. After Marr [1982] p.25

theory" and "Representation and algorithm" in table 1.1. Then we review several state-of-the-art methods from the literature used to implement computer vision applications. We analyze the role that each of them could have in the scope of our framework.

The PhD Thesis has been motivated by concerns arising from the development of real world applications. We have selected four different types of application to test the principles of our theory. The objective is not to find the best method solving each problem, but to evaluate whether the application of these principles leads to results comparable to human vision in a variety of applications, and therefore evaluate their validity.

The dissertation is structured in five chapters, as follows:

- Chapter 1 introduces the topic of visual perception and gives the motivations, outline and contributions of this PhD Thesis.
- Chapter 2 reviews works related to visual perception from the fields of neurophysiology and psychology, so that human vision results are better understood.
- Chapter 3 presents a novel framework for machine visual perception. We follow Marr [1982] scheme with three levels of explanation. We first describe

a computational theory for vision, then a representation and algorithm, and finally we review state-of-the-art methods to implement the fundamental operations of the algorithm.

- Chapter 4 studies four applications of computer vision with different types of perception: perception of activity, authorship, intrusion and aesthetics. Human experts would likely suggest approaches based on high level features, but in all the cases results comparable to those given by human vision can be achieved without following the human-based suggestions.

- Chapter 5 concludes the dissertation summarizing the main results obtained and outlining future research.

1.5 Research contributions

The research related with this PhD Dissertation yield the following contributions:

1.5.1 Articles

1.5.1.1 Learning crowd behavior for event recognition

This paper presents a new method for event recognition based on machine learning techniques. One machine is trained per kind of event using color, texture and shape features. Testing is performed on the PETS 2009 dataset. We evaluate accuracy of our automatic system with six different kind of events and then compare the results with human classification (Cermeño *et al.* [2013]).

1.5.1.2 Offline handwriting segmentation for writer identification

In this paper we present a new technique for off-line text-independent handwriting analysis based on segmentation. Segmentation is a common step used in different research works in order to generate connected components that will be processed to extract features (geometry, concavity etc.). Our work focuses in the segmentation process and the information that can be directly extracted from the way a writer joins or separates ink connected components without need of analyzing the

components themselves. The proposed multi-segmentation method shows good results tested on its own with real documents from police corps database and suggest an improved way to apply segmentation to other connected component based systems (Cermeño *et al.* [2014a]).

1.5.1.3 Simulation of human opinions about calligraphy aesthetic

This paper proposes a method for simulating human opinions about graphical artistic expressions like calligraphy using computers. Scanned images of handwriting texts from a large database are labeled as "beautiful writing" or "ugly writing" by two persons based on their own likes. Our objective is to replicate these opinions using machine learning techniques. Shape features are extracted from the images in order to encode aesthetic principles. A classifier based on k-nearest-neighbors algorithm is trained to automatically label images. The results are promising since most of the different configurations of the system present good performance. Both, method and feature selection results could be of use for future work on aesthetic classification by computers (Perez *et al.* [2014]).

1.5.1.4 Intelligent video surveillance beyond robust background modeling

The increasing number of video surveillance cameras is challenging video control systems. Different video analysis methods have effectively met the main requirements from the industry of perimeter protection. High accuracy detection systems are able to process real time video on affordable hardware. However some problematic environments cause a massive number of false alerts. Many approaches in the literature do not consider this kind of environments while others use metrics that dilute their impact on results. A video surveillance solution implemented as an intruder detector will repeat steadily the same false alerts and can hardly be considered to be "intelligent". We benefit from the observation that problematic environments only occur occasionally to propose a method that manages directly these environments when they show up. Our approach is based on machine learning and global features, bringing adaptability to the video surveillance solution. Tests with thousands of hours of video show how good an

intruder detector can perform but also how a simple fault in a camera can flood a control center with alerts. The new proposal is able to learn and recognize events such that alerts from problematic environments can be properly handled (Cermeño *et al.* [2017b]).

1.5.1.5 Segmentation as a characteristic for writer identification

Forensic experts are able to identify the authorship of a document by analyzing its handwriting. Computer vision methods have been used to automatize this task. However, like for many other computer vision applications, segmentation represents a problem. The challenge is to segment words into characters, such that pattern recognition techniques can be used to classify them. Character classification has proved to be a successful approach but automatic segmentation very often shows poor results. In this work we show how segmentation can by itself help to identify writers. Segmenting handwriting into connected components is a simple and common step in writer identification methods, however those with better results usually require to segment connected components into smaller units. We propose a new framework for handwriting segmentation, in which instead of using multiple segmentation techniques, we use several values for a segmentation parameter. Our method is only based in connected components and correctly identifies 92% of the authors of free-style handwritten documents (Cermeño *et al.* [2014b]).

1.5.2 Patents

1.5.2.1 Method and device for change in illumination for vision systems

The invention relates to a method and device for the detection of changes in illumination for vision systems between a digital image of an area and a digital image of a background model of said region of study of the same size, wherein, based on such images, at least one blob of a region which reflects the differences between the background model and the current or detection image of the area is selected by segmentation techniques, the spatial correlation between the pixels of

the blob in the detection image and in the background model image being found. According to the correlation with respect to a threshold value thereof, this change is associated with a change in illumination. An implementation suitable for video surveillance systems is provided together with the previous device (Gonzalez *et al.* [2017]).

1.5.2.2 Video surveillance system based on the analysis of sequences of images generated by events

This invention is framed in the field of video surveillance, that is, of activity or presence detection technologies based on video analysis. More specifically, the invention relates to a video surveillance system configured to perform precision analysis of the captured images, wherein the analysis is applied to certain sequences of images or clips defined by different configurable events. The system described in the invention further allows its combination with other traditional video detection and processing systems, substantially improving their efficiency and accuracy (Cermeño *et al.* [2017a]).

Chapter 2

Related works

2.1 Physiological basis of vision

Before talking about pixels, frames, video or computing we will review some concepts of the physiological basis of visual perception. Physiology is the branch of biology dealing with the functions and activities of living organisms and their parts, including all physical and chemical processes. Trying to understand the processes behind the transformation of a subset of electromagnetic radiation (light) into information treatable by our brain will give us ideas or a reference for a better achievement of our work. In the past, neurophysiology has successfully inspired some of most important methods in pattern recognition (LeCun *et al.* [2015]).

Our task has been greatly supported by excellent publications that either introduce or bring together the most important findings from biologists and neurophysiologists in the field of visual perception. Yantis [2001] has collected some of the best articles written about vision in the book "Visual Perception: Essential reading". "Basic Vision: an introduction to visual perception" (Snowden *et al.* [2006]) is an enjoyable and easy to read book full of practical images that let the reader experience some of the visual phenomena (ex: Troxler fading). "Visual Perception: Physiology, Psychology and Ecology" (Bruce & Green [1990]) is the main reference for both most of the section and of the present chapter. Yantis [2001] and Snowden *et al.* [2006] also cover ground for other sections, but in Bruce & Green [1990] we found many of ideas for the approach we were looking for.

2.1.1 Light

We will introduce the concept of vision along with the definition of light. Light is one form of energy that is reflected or emitted from objects in the form of electrical and magnetic waves that can travel through space with a wavelenght of 400 to 700 nm 2.1, which is the bandwidth perceived by most people. We could say that light is defined by means of the limitations of the human visual system.

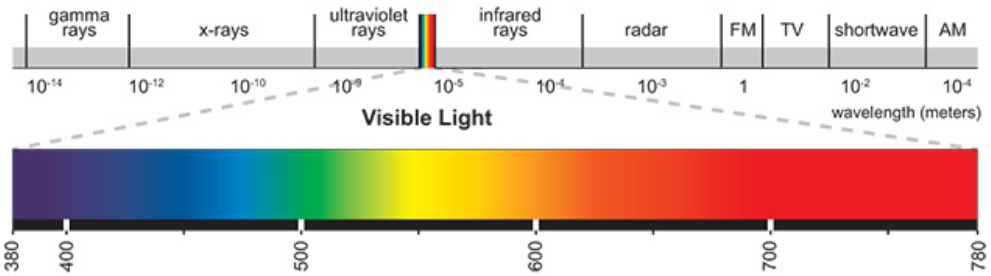


Figure 2.1: Light spectrum

Objects, animals or people emit an electromagnetic radiation called thermal radiation if their temperature is above absolut zero. Most of the times we do not see this radiation because its wavelenght (14.000 nm or more) is far away from the visible bandwidth. If you heat up an object its themal radiation changes. Around 798K a solid or liquid starts to glow with a middly dull red color. At higher temperatures the substance becomes brighter and its color changes from red towards white and finally blue. Incandescent light bulbs or lamps are an example of every day use of this reaction. A wire filament is heated to a high temperature (2500K or more) until it glows emiting visible light (visible is redundant when used with light but we may use it to avoid confusions). A detailed description about the relation between heat and light can be found in the classic book Wickenden [1910].

Objects around us are not at 2500K but 270K to 315K (-3 to 42 degrees celsius), their thermal radiation is out of the visible bandwidth, so it is not that kind of energy that we usually perceive but instead we usually perceive the light reflected by matter.

When light passes through a medium, even transparent (e.g.: air or water),

photons collide with particles of matter giving up their energy and disappearing in a process called absorption. Absorption is different in water (stronger) than in air. Longer wavelenghts are absorbed more strongly, making deeper water progressively bluer. If light passes through a transparent or translucent medium its energy is not absorbed but as a result of the change of medium rays may change their direction in a phenomena called refraction. When light reaches an opaque surface, some of its energy is absorbed and some of it is reflected. Black surfaces for example absorb most of the light falling on it and reflect little. Silvery surfaces do the opposite and reflect most of the light.

In general surfaces change the spectral composition of the light reflected from them by absorbing some wavelenghts more strongly than others. The texture of a surface has an important effect on how coherently light is reflected. Smooth surfaces reflect light uniformly while rougher ones will reflect light with different angles in an incoherent way.

2.1.2 The eye

The organ in charge of catching and converting light energy into neural signals is the eye. The main elements of the eye are pupil, iris, cornea, lens and retina. Pupil is the aperture of the eye allowing light to strike the retina. The iris is a circular structure responsible for controlling the diameter and size of the pupil thus the amount of light that reaches the retina. The cornea is the transparent front part of the eye that covers iris, pupil and aqueous humour. The lens is also a transparent structure that along with the cornea helps to refract light to be focused on the retina. It is clear that the retina plays a key role in the transduction of light, so in this section we will focus on it.

The retina is a complex structure composed of several layers of neurons, some of which are sensitive to light, the photoreceptors cells. There are mainly two types of photoreceptors in the human eye, rods and cones. Rods respond very well to extremely dim light and are therefore very useful in dim conditions (night). When the rods are exposed to high levels of light for a prolonged period they become desensitized because of saturation. The rod system, also called scotopic, is useless in full daylight.

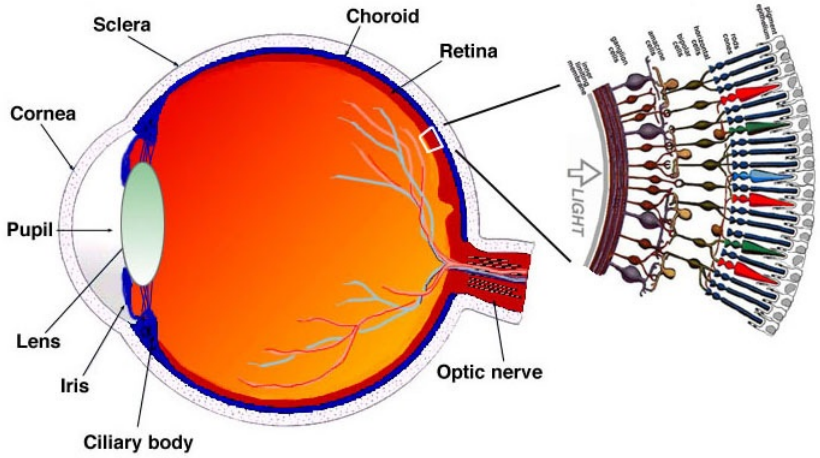


Figure 2.2: Human eye with a schematic enlargement of the retina (http:// webvision.med.utah.edu/)

On the other hand the cone (or photopic) system operates best at greater light intensities. The photopic system is made of three types of cones. The first type sometimes called “red” or “L” cones are more sensitive to long wavelenghts, the second type called “green” or “M” are most sensitive to middle wavelenghts while “blue” or “S” cones are more sensitive to shorter wavelenghts. Our color vision is possible thanks to these different types of cones.

Together scotopic and photopic systems enable us to detect lights that differ in amount by many orders of magnitude. Human vision is able to handle approximately seven log units of light intensity, but at any one time it is effective over a range of only one or two log units. Our visual system can adapt itself to cover higher or lower ranges in a process called light adaptation. When light intensity suddenly increases, photoreceptors impulses rise rapidly and then fall to a steady level. When a photoreceptor is adapted to light and then left in darkness its sensitivity to light gradually rises. The process of dark adaptation is much slower than light adaptation. As a result of these processes the output of photoreceptors is quite stable over a wide range of light intensities. Sudden changes in light intensity like an object occluding a light source, a shadow etc.

will activate the photoreceptors. This way gradual changes in the environment such as diurnal fluctuation in light level are somehow filtered while events that could be relevant (prey or predator appearance) have a clear impact in the neural activity of the retina. The retina, visual adaptation and photoreceptors are well covered in Dowling [1987].

Hartline *et al.* [1956] demonstrates that the responses of the photoreceptors interact with one another through a process of lateral inhibition. Each cell inhibits the firing rate of those in a roughly circular area around it, thus enhancing the contrast in light patterns and sharpening the perception of shapes. Lateral inhibition stands out rapid spatial changes, very much like adaptation gives prominence to temporal changes in light intensities.

Another layer of neurons in the retina is made of cells of different sizes called retinal ganglion cells. Larger ones are refered to as M cells (M for magnocellular) while smaller ones are called P cells (P for parvocellular). Ganglion cells receive information from photoreceptors which synapse with it. P cells activity depends on the wavelength of the light reaching the retina, integrating the information from the red and green cones. For example some P cells are excited by green cones and inhibited by red cones. On the other hand M cells seem to mix the signals from different cones (Snowden *et al.* [2006]). It seems that P neurons in the retina are concerned with color while M neurons process information about motion. We will get back to this idea in the next section.

Physiological measurements on the ganglion cells show that a spot of light shining on a small part of the retina will only modify the activity of a few ganglion cells while others present no changes. We define the receptive field of a neuron in the visual system as the area on the retina over which light stimulus can modify its behaviour. Kuffler [1953] shows that the effect of a spot of light on a receptive field depends on whether the light falls in a small circular area in the center of the field or in the surrounding area. There are two types of this receptive field, called "ON-centre" and "OFF-centre". Snowden *et al.* [2006] defines an ON-centre unit as one whose firing rate increases when light hits its centre and decreases when light hits the surrounding, and an OFF-centre unit as one whose firing rate decreases when light hits its centre and increases when light hits the surrounding (figure 2.3).

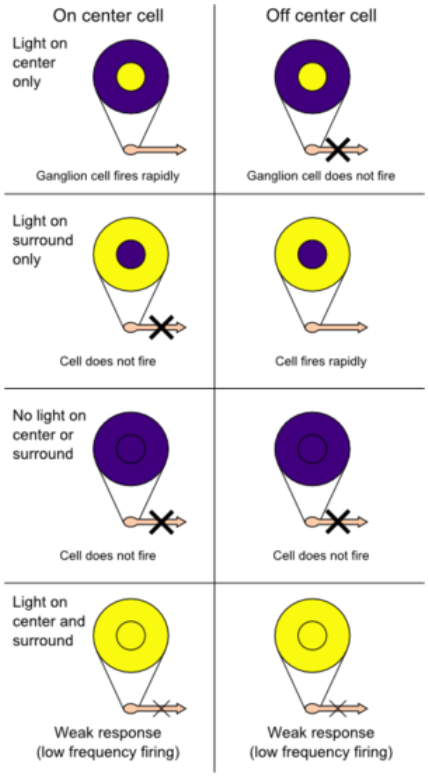


Figure 2.3: On and Off center receptive fields respond to light stimulus (Wikimedia Commons)

What happens if we stimulate both center and surrounding? Together these inputs tend to cancel out so there is is little or no change in the response of the cell. In order to get a response from a cell we need to have a change in the light occurring within the receptive field. Retinal ganglion cells only respond if there are changes in the luminance profile within the receptive field, such as an edge and do not respond to changes in the overall luminance of the whole visual field.

2.1.3 Visual pathways

Ganglion cell axons compose the optic nerve which is in charge to transmit the visual information from the retina to the brain. The optic nerve from one eye leaves it at the blind spot and converge with the optic nerve from the other eye in the optic chiasm. At this stage some fibers cross over to the other side of the

brain and the optic nerve changes its name to optic tract.

The optic tract reaches the thalamus that among other functions relays sensory information from the retina to the cerebral cortex. With respect to the visual system the relevant relay is the lateral geniculate nucleus (LGN). LGN has a distinctive structure made of six layers of neurons, three receive input from one eye and the other three from the other. There are two magnocellular layers and four parvocellular layers. The magno layers are contacted by the axons of the M cells while parvo layers are contacted by P cells from the retina (Purves & Lotto [2003]). This way of routing information from the retina to the LGN defines two pathways, sometimes referred to as M and P pathways.

Livingstone & Hubel [1987] presents evidences that cells in the P and M pathways handle different visual information such as color properties, contrast sensitivity, spatial resolution and temporal properties. For example "over 80% of P neurons show color-opponency... while M neurons receive summating input from the red and green cones". The P pathway seems to be more suited for encoding color. M cells are more sensitive than P cells to luminance contrast and have a shorter latency. The M pathway seems to be more suited to process information about changes in the stimulus, like motion or flickering. The existence of these differentiated pathways suggests some kind of parallel processing in the visual system. Stone [2013] treats this topic in depth.

Another interesting feature of the visual pathways is the retinopic mapping of stimulus. Light from two adjacent parts of the visual world reflect on to adjacent segments of photoreceptors in the retina, that project into adjacent ganglion cells that are connected with adjacent LGN cells forming an orderly map of the visual world. LGN cells have, like retinal ganglion cells, concentric receptive fields (Bruce & Green [1990] p.49-50).

On the other hand, there is an important difference between the retinal ganglion and the LGN cells. The ones in the LGN receive its biggest input from the cortex, the area where LGN sends its output. So the biggest input to the LGN comes "top-down" rather than "bottom-up". This has led to the idea that the LGN might be important in filtering what information gets through to the cortex (Snowden *et al.* [2006] p.39).

The primary visual cortex, also called striate cortex or just V1 is the largest

of the visual areas and clearly very important for vision. Here again we find that the retinopic mapping from LGN is present. Things that are close together in the visual scene are imaged on neighboring areas of the retina. They will be processed by neighboring cells in the LGN and will be analyzed by neighboring neurons of the visual cortex.

Hubel & Wiesel [1959] describes two classes of neurons in the cortex with receptive fields behaving in a different way than the ones we saw before: simple and complex cells. Simple cells perform linear spatial summation of light intensity in their fields. Their responses to stationary patterns of light depend on the position and orientation of the stimulus. They present their maximum response to a bar or edge oriented at a particular angle to the visual axis 2.4.

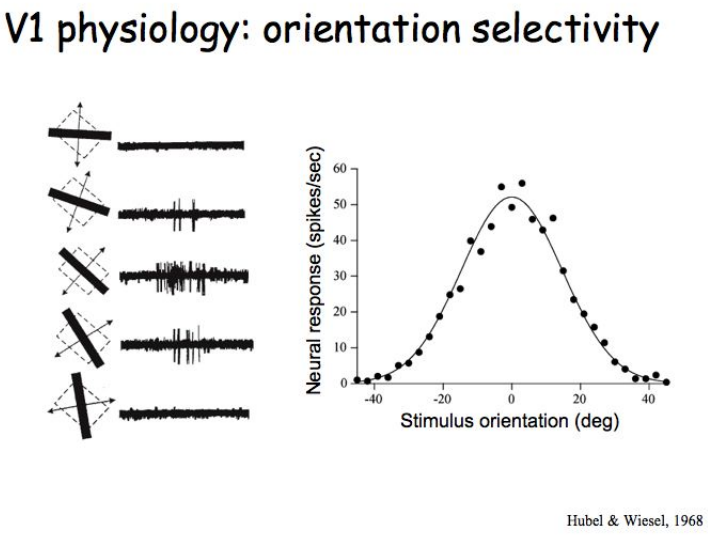


Figure 2.4: Simple cell response to different orientations of a stimuli after Hubel & Wiesel [1968]

Complex cells are also more responsive to orientated lines but do not show discreate ON and OFF regions, so if a small spot of light strikes a point in the receptive field the cell may give both ON and OFF output. Hubel and Wiesel proposed that complex cells could be built with a combination of simpe cells with connected outputs. The complex cell could operate with these outputs, for example an OR operation. A third class of cell described by the authors are the

hypercomplex cells, which are also sensitive to the size of the bar. If these cells belong to a third class or are just a subtype of complex cells has been argued and even the authors of Hubel & Wiesel [2005] correct themselves for some of their assumptions from 1959.

In Hubel & Wiesel [1962] visual pathways are presented as a continuous process with several levels where each level requires more parameters in order to influence the firing of its cells. This hierarchical organization begins in the retina where a spot of light with correct position, size and intensity will make a ganglion cell fire. Geniculate cells appear to be more sensitive to the size of the spot, demanding something closer to the optimum to fire. When the information from the LGN reaches the simple cells from the cortex requirements are increased and a specific orientation is necessary.

When we reach complex cells, refered to as "higher-order neurons" by the authors, responses become less selective. Complex cells may be concerned with stimulus orientation but they may not be so demanding towards the stimulus position: "Their responsiveness to the abstraction that we call orientation is thus generalized over a considerable retinal area" (Hubel & Wiesel [1962]).

Through this review of works on how visual stimulus are treated by neurons from the retina to the cortex we found several interesting concepts like "day and night" systems, adaptation , edge detectors, parallel processing, hierarchical organization, specialization or generalization. We mentioned LGN or V1, but "forgot" about many other areas of the brain that deal with vision (superior culliculus, V2, V3 etc). We consider that traveling further on the visual pathway is out of the scope of this work. Actually a complete physiological description of vision is still missing, even if a lot has been, and is being done to better understand how our brain works.

2.2 Theories about visual perception

> Trying to understand perception by studying only neurons is like trying to understand bird flight by studying only feathers: it just cannot be done. In order to study bird flight we have to understand aerodynamics; only then do the structure of feathers and the different shapes

of bird wings make sense (Marr [1982] p.27).

Marr's image of bird flight stands out the difficulty to understand vision only through the analysis of neural activity. In this section we will present some of the most important theories and hypothesis about visual perception. Some of them do try to fill the gap between our subjective perceptions and the activity of our neurons, whereas others give more importance to different psychological hypotheses. Reading arguments defending one theory and rejecting others has been very instructive. In this section we have tried to present the main ideas and concepts from some of the most relevant scientists that have worked in the field of perception. This does not mean that we agree or disagree with them, just that we find them interesting to define a theory for machine visual perception.

2.2.1 Single neuron hypothesis

Barlow [1972] pushes further the idea of neuron specialisation and proposes that "our perceptions are caused by the activity of a rather small number of neurons selected from a very large population of predominantly silent cells. The activity of each single cell is thus an important perceptual event". In a previous work with frogs Barlow [1953] shows that a black disc moving rapidly to and fro within the receptive field of one particular type of ganglion cell caused a strong response that could be maintained as long as the movement was continued. When the same stimulus was presented to an intact frog, there was a sudden reaction of a jump and snap. This reaction suggests that Barlow had found a neuron behaving as a "bug detector", and that this "bug detector" is directly a retinal cell, not a "higher-order" cell in the cortex. Lettvin *et al.* [1959] suggests that actually there are four different classes of specialised neurons in the frog's retina whose activation is nearly independent of the general illumination: contrast, convexity, movement and dimming detectors. The second class of neuron "convexity detector" has the same behavior that the "bug detector" described by Barlow [1953].

In general, Barlow [1972] considers neuron activities as thought processes, able to discriminate depth of objects, ignore irrelevant causes of variation, give prominence to what is relevant or detect patterns. Perret *et al.* [1982] presents the response of cells in the superior temporal sulcus (STS) to face patterns, either

real, projected, human or rhesus monkey faces. Among 497 neurones, 48 seemed to be activated by faces, since the response to face patterns was two to ten times larger than the response to gratings, simple geometrical stimuli or complex 3D objects. Barlow [1969] uses the concept of "trigger features" to refer to specific stimuli that make a cell fire.

In order to get a deeper understanding on how these neurons react to face features, Perret *et al.* [1982] covers some parts of the faces or presented parts isolated (eyes, mouth, hair etc.). The results show that some cells were activated by features of the face as well as they were by the whole face. Different cells were activated by different features, and combined features had stronger responses than any of them tested individually. The authors suggest that they represent a high stage in visual processing of faces, and stick to the theory that complex patterns can be coded at a single cell level.

Sherrington [1941] uses the notion of "one ultimate pontificial nerve-cell, ... the climax of the whole system integration" in opposition to the notion of mind as "million-fold democracy whose each unit is a cell", which he believed was more accurate. However the "pontificial cell" is an interesting concept for us. Barlow [1972] proposes that it should be replaced by "cardinal cells" because "the whole of subjective experience at any one time must correspond to a specific combination of active cells. Among all the cardinals only a few speak at once". The concept of "pontificial cell" means that for every object or scene that can be recognized there must be a single cell specialised to do it. Since our perception usually includes several objects or scenes, several cells would be required, thus the "cardinal cells". We could easily find the parallelism between "cardinal cells" and the group of specialised neurons involved in the visual processing of faces described by Perret *et al.* [1982].

The "pontificial cell" concept has several weaknesses, for example the amount of possible perceptions is probably larger than the number of neurons in the brain. A second problem would be the variations of some percepts, for example people faces. Lettvin used "grandmother cell" to refer to a hypothetical cell that would be able to recognise all views of grandmother's face. This hypothetical cell should be able to cope with different poses, hair-dresses, glasses, age effects etc. A group of cells sensitive to features, like the ones described by Perret *et al.* [1982], or even

a group of "cardinal cells" able to learn the different percepts the grandmother's face could create, are likely better to succced in the task of recognition than just a "grandmother cell". Quiroga *et al.* [2008] rejects the idea of the "grandmother cell" and suggest a very sparse representation of information.

Quiroga *et al.* [2005] reports neurons that are selectively activated by different pictures of people, landmarks, objects or letter strings. For example they found a neuron that fires to pictures of the Eiffel Tower and Tower of Pisa but not to other landmarks, or another cell that fires to Jennifer Aniston and Lisa Kudrow pictures, both actresses in TV series "Friends". In the words of the authors "results suggest an invariant, sparse and explicit code, which might be important in the transformation of complex visual percepts into long-term and more abstract memories."

2.2.2 Computational theory of vision

"The transformation of complex visual percepts into long-term and more abstract memories" could be handled by "a complex information-processing system". The last quotation is the title of the second section in the first chapter of Marr [1982], which approaches vision as a complex information-processing task that "produces from images of the external world a description that is useful to the viewer and not cluttered with irrelevant information" (Marr [1982] p.31). Marr is the father of the theory we present in this section, and we are lucky to have most of his research collected and clearly explained in his book Vision. That is why we will continuously refer to it through all the section.

Marr proposes that understanding vision requires questions like "why" and "how". To him, scientists in the 1950s and 1960s put most of the efforts in describing the behavior of cells. Barlow [1953] presents neurons that behave like bug detectors, Perret *et al.* [1982] describes neurons that look like face detectors or Gross *et al.* [1972] who shows neurons that could play the role of hand detectors, but these findings would not explain "why or even how such a thing [detector] may be constructed from the outputs of previously discovered cells" (Marr [1982] p.15).

According to Marr [1982] p.19:

> there must exist an additional level of understanding at which the character of the information-processing tasks carried out during perception are analyzed and understood in a way that is independent of the particular mechanisms and structures that implement them in our heads. This was what was missing - the analysis of the problem as an information-processing task. Such analysis does not usurp an understanding at the other levels - of neuron or of computer programs - but it is a necessary complement to them, since without it there can be no real understanding of the function of all those neurons.

If we replace "neurons" by "computer vision techniques" we find one of the motivations of our thesis (1.2). Without a high-level analysis of the problem of perception we cannot really understand why we should use such or such technique to build computer vision systems.

Marr introduces a "new level" of analysis of the "problem": the computation theory level, which is a complement to other levels: representation and algorithm, and hardware implementation. The original definition of the three levels is included in figure 1.1. In these terms, neural activity description would fall in the "hardware implementation" level. Neurophysiology research usually would be related to this level, however some findings could help to understand the type of representation being used. The author underlines the importance to "have a clear idea about what information needs to be represented and what processes need to be implemented" before making inferences from neurophysiological findings about algorithms and representations.

The concepts "representation" and "process" have several pages dedicated in Marr [1982] p.20-24. Representation is important to solve problems, Marr uses the example of roman and arabic numeral systems which both represent numbers. The effort required to multiply arabic numbers (e.g.: 1240 x 349) is lower than the one required to multiply roman numbers (XXVII x CXXI). We will discuss about representation in next chapters, since it is indeed a big issue for any computer program and even bigger for one related with machine vision. An algorithm requires representation for its input and for the output, which can be the same for both, or not.

Processes in visual perception must derive properties of the world from images

of it. A pattern of light could be a representation of the visual world in a particular moment. The neural response from the photoreceptors in the retina could be another one. The transformation of light into nerve impulses is a process where an electromagnetic representation of a scene is transformed into an electrical one. But as we mentioned before, for Marr the key point, and also the missing point, was the top level, a computational theory for vision, since "the computations that underlie perception depends more upon the computational problems that have to be solved than upon the particular hardware in which their solutions are implemented" (p.27).

So, what is the "computational problems that we have to solve" in human vision? One of the best answers we find in the book could be "building a description of the shapes and positions of things from images" (p.36). We underline "human vision" because Marr judges his approach good enough to understand vision from different animals, not only humans. The main idea is that vision is used for different purposes by the different animals, so different representations and processes would be necessary to understand their vision. For example, Marr suggests that a housefly may not have "any explicit representation of the visual world around him- no true conception of a surface, for example, but just a few triggers and some specifically fly-centered parameters" (p.34). Humans do have an explicit representation of the visual world, and thus need vision to give a description of shapes and positions of the things from images.

Most chapters of the book (2 to 5) describe in detail a representational framework for deriving shape from images and answer the question Marr considers necessary to understand human vision completely:

> what kind of information does the human visual system represent, what kind of computations does it perform to obtain this information, and why? How does it represent this information, and how are the real computations performed and with what algorithms? Once these questions have been answered, we can finally ask, how are these specific representations and algorithms implemented in neural machinery? (Marr [1982] p.99)

In the next chapter (3) we will propose answers to all these questions for

machine visual perception systems.

Table 2.1 summarizes the overall framework. We will not get into a detailed description of all the processes and represention introduced by Marr [1982]. However we want to present some extra relevant concepts treated by the author. The first one is modularity. Every computer scientist knows that a large problem should be addresed by solving smaller problems. If these smaller problems can be solved by independant modules, the result would be easier to debug and improve. Marr [1982] (p.102) enumerates these advantages and incorporates "the principle of modular design" to the processes described for vision understanding: "The existence of modular organization in the human visual processing proves that different types of information can be analyzed in relative isolation". Following this principle Marr presents computational theories for different decoding processes like stereopsis, directional selectivity, structure from apparent motion, depth from optical flow, surface orientation from surface contours, surface orientation from surface texture, shape from shading, photometric stereo or light and color as an approximation to reflectance.

The principle of modularity is applied in the different stages of the representational framework (2.1), for example in the 3D model representation (p.313). According to the author "recognition involves two things: a collection of stored 3D model descriptions and various indexes into the collection that allow a newly derived description to be associated with a description in the collection" (p. 318). In his theory there are three indexes. With the first one, called specific, 3D models without component decomposition are matched, then, in the next level details about components are required to finally find the correct 3D model. The adjunct index provides access to 3D models for its components based on their locations, orientations and relative sizes. The third index, called parent index uses component recognition to recognize the whole shape (ex: recognizing horse legs provides access to horse shape).

The last point we wish to outline are "constraints" or rules. When Marr introduces the concept "computational theory" he states that

> its important features are: (1) that it contains separate arguments about what is computed and why, and (2) that the resulting operation

Name	Purpose	Primitives
Image(s)	Represents intensity	Intensity value at each point in the image
Primal sketch	Makes explicit important information about the two-dimensional image, primarily the intensity changes there and their geometrical distribution and organization.	Zero-crossing Blobs Terminations and discontinuities Edge segments Virtual lines Groups Curvilinear organization Boundaries
$2^1/_2$-D sketch	Makes explicit the orientation and rough depth of the visible surfaces, and contours of discontinuities in these quantities in a viewer-centered coordinate frame.	Local surface orientation (the "needles" primitives) Distance from viewer Discontinuities in depth Discontinuities in surface orientation
3-D model representation	Describes shapes and their spatial organization in an object-centered coordinate frame, using a modular hierarchical representation that includes volumetric primitives(i.e., primitives that represent the volume of space that a shape occupies) as well as surface primitives.	3-D models arranged hierarchically, each one based on a spatial configuration of a few sticks or axes, to which volumetric or surface shape primitives are attached.

Table 2.1: Representational framework for deriving shape based on the original table by Marr [1982]

> is defined uniquely by the constraints it has to satisfy. In the theory of visual processes, the underlying task is to reliably derive properties of the world from images of it; the business of isolating constraints that are both powerful enough to allow a process to be defined and generally true of the world is a central theme of our inquiry (Marr [1982] p.23).

The first part of the quotation "resulting operation is defined uniquely by the constraints it has to satisfy" introduces the idea of constraint-based programming, which is "to solve problems by simply stating constraints (conditions, properties) which must be satisfied by a solution of the problem" (Fruhwirth & Abdennadher [2003] p.2). Representations and processes are designed by constraints and assumptions (Marr [1982] p.43 & p.267). It seems that the whole "problem" of vision could be addressed by means of a modular constraint solving approach.

2.2.3 The ecological approach to visual perception

> In perception, perhaps the nearest anyone came to the level of computational theory was Gibson (Marr [1982] p.29).

> The theory of information pickup purports to be an alternative to the traditional theories of perception. It differs from them all (Gibson [1986] p.251).

In this section we will present the "Ecological approach to visual perception" introduced by Gibson and explained in detail in his book "The Ecological Approach to Visual Perception" originally published in 1979. For this study we have used a latter publication: Gibson [1986]. Our introductory paragraph shows how Marr and Gibson did not agree even in their disagreement. Marr considers the ecological approach to be very close to the computational theory while Gibson strongly rejects any information processing based theory. Some works like Bruce & Green [1990] suggest that the ecological approach could be considered at a more global level of analysis than the computational. From the point of view of machine perception, Gibson's works provide useful ideas that perfectly fit within Marr's framework.

For Gibson "perceiving is an achievement of the individual, not an appearance in the theater of his conciousness" (Gibson [1986] p.239). He rejects the idea that vision is based on processes like recognition, interpretation, storage or retrieval of ideas, applied over an image (representation) in the brain. According to Gibson we do not perceive color, form, location, time or motion (Gibson [1986] p.85), when we "see" objects, places or events, we are perceiving what these things afford, the so called "affordances". A house could afford "sleeping" or "staying warm", an apple "eating" or "throwing". An affordance is the opportunity for action provided by a particular object or environment. But how do we perceive these affordances? In Gibson's approach this is done by continuously picking up information from the ambient optic array, that is the structure arrangement of light with respect to a point of observation. In other words, the spatial pattern of light reflected by textures from different surfaces. To better define the concept of picking up information we use Gibson's own words.

> Picking up information is not to be thought of as a case of communicating. The world does not speak to the observer. Animals and humans communicate with cries, gestures, speech, pictures, writing and television but we cannot hope to undertand perception in terms of these channels; it is quite the other way around. Words and pictures convey infomation, carry it or transmit it, but the information in the sea of energy around each of us, luminous or mechanical or chemical energy is not conveyed. It is simply there. The assumption that information can be transmitted and the assumption that it can be stored are appropriate for the theory of communication, not for the theory of perception (Gibson [1986] p.242).

This description attacks other theories' basis, starting with the concept of information. If information cannot be transmitted or stored the different representations described by Marr would not make sense. Gibson's theory states that we do not have to process any kind of information because the information is already there, in the structure of light. When we "perceive" we directly pick up this information. Gibson uses the word information to refer to a "specification of

the observers's environment not to the specification of the observers's receptors or sense organs" (p.242).

Instead of a sense, the theory of information pickup requires a perceptual system. Gibson underlines that a perceptual system is active whereas a sense is passive. A sense has receptors, a perceptual system is made of organs, in the case of the perceptual system, it would include lens, pupil, chamber, retina in the first level, eye muscles, mobile head etc. in the following levels up to the body itself, whose movements change the optic array. The perceptual system can orient, explore, investigate, adjust, optimize, resonate, extract and come to an equilibrium.

Several interesting ideas follow the "active" character of a perceptual system. In the case of senses, attention is something that can be consciously focused, while in the perceptual system it is a skill than can be educated (p.246). We could think about examples for this assertion. Why do people that study, repair or sell some kind of product (e.g.: a toaster) usually notice its presence while other people do not? Even if both can recognize the product without problems. It can be observed that the perceptual system of the one working with the product is more educated to perceive it. In the words of Gibson, the perceptual system "has become sensitized", this happens when it is attuned to a certain sort of information.

We already mentioned that a perceptual system can be adjusted, optimized or attuned. It does not require memory. Gibson rejects the idea of memory as the bridge between the past and the present, the assumption that past ceases to exist unless it is preserved in memory, or the existence of images or pictures representing the past stored somewhere. Instead of storing images of the past, Gibson proposes that recognition is done thanks to a perceptual system that resonates to invariants of the structure of light. These propositions have been successfully implemented with ANN (Duda *et al.* [2012]), patterns can be stored by modifying the parameters of an ANN.

The information a child uses to identify his mother despite the different figures she may have in the optic array are features of her (eyes, mouth, hair etc.) that are invariant to a certain degree. In the ecological approach invariants refer to some measure of the structure of light reflected from an object, a scene or an event that

remains constant while other measures vary. In Gibson's words, invariants specify the persistence of the environment. According to him everything in the world persists in some respects and changes in others. In order to perceive persistence and change we pick up invariants of the structure of the ambient optic array. The concept of invariance is somehow loose. There are invariants specifying every particular face, landscape, painting, animal, place etc. In fact, Gibson's definition of abstraction is "invariance detection across objects" (p.249). As we pointed before (1.2) finding robust features has been one of the tasks, that has attracted more attention in the field of image recognition. In fact results suggest that such invariants can successfully be extracted (Simonyan & Zisserman [2015]).

The theory of picking up information offers more interesting ideas. For example, information in the ambient light is inexhaustible. "A perceiver can keep on noticing facts about the world she lives in to the end of her life without ever reaching a limit" (p.243). We think it should be easy for any reader to find his own example of noticing something that has been there for years, just in front of his eyes. Our perception changes, not only because of changes in the environment but also in ourselves. The same object may be perceived differently depending on factors such as necessity. The ecological approach to vision states that all the information is there and we continuously are picking up parts of it. Even if the same information is available two different persons will pick up different parts. This is a nice introduction to the fact that different people have different perceptions in the same situation. Using the ecological approach terms, different people may have perceptual systems that have been attuned differently and resonate in different ways to the invariants of the optic array.

2.2.4 Gestalt laws of perceptual organization

Invariance is the main difference between Gibson's concept of affordance and the concept from which it derives, Koffka's "demand character".

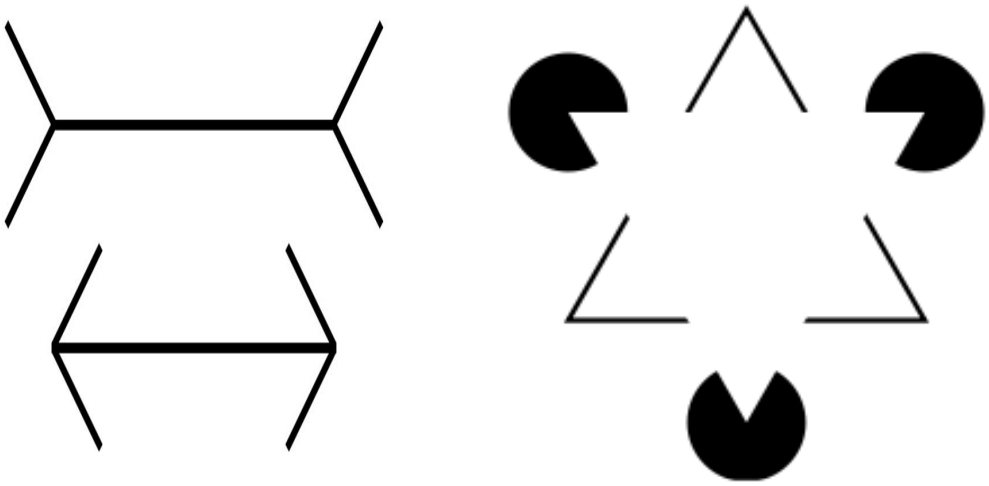
> The post-box has a demand character only when the observer needs to mail a letter. He is attracted to it when he has to post, not otherwise. The value of something was assumed to change as the need of the observer changed (Gibson [1986] p.138-139).

Contrary to "demand character", affordances are invariant so they are always there, even if the observer does not perceive them. In this section we will leave Gibson's terms and focus on the Gestalt psychologists' ideas, who not only influenced him but many others, including Marr.

Not surprisingly, we will start by one of the "founders" of the Gestalt school: Koffka. Koffka [1935] presents the idea of a prescientific stage where man would behave as the situation tells him to behave: "fruit says 'Eat me'; water says 'Drink me'; thunder says, 'Fear me' and woman says, 'Love me'" (p.7). According to Koffka man has learned to "distrust what things told him" and find the errors in his original world based on knowledge of individual things. He opposes this knowledge to the scientific knowledge resulting from a new activity called thinking. For the moment we will look to the ideas of "error" and "distrust", avoiding other interesting considerations that arise from "direct knowledge" versus "scientific knowledge". German and romance languages offer different words for each type of knowledge. Direct knowledge could be associated with "Cognitionis" in latin and "Kenntnis" in German, while scientific knowledge could be "Sapientae" in latin and "Wissen" in German.

A false or misleading perception is called illusion. Figure 2.5 includes two examples. In the first case (a) we have two lines with objectively the same length but one appear to be longer that the other. In the second case (b) we perceive a white triangle that does not exist.

As we can see, sometimes, our perception is wrong, the information catched by our visual system does not correspond to reality. Other times our visual system can get more than one perception from the same stimulus, a well known example is E. Rubin's vase (figure 2.6). The picture can be seen either as a pair of black faces over a white background or as a white vase over a black background, but both cannot be perceived at the same time. In this picture the definition of "ground" and "figure" is ambiguous. The work about figure and ground Rubin [1958] proposes that a common border of two fields determines the perception of a figure and background, being the figure the field more affected by the "shaping effect". According to Rubin fields experienced as figures are richer, with a more differentiated structure, with greater structural solidity of the color and appear to be closer to the viewer than the field experienced as background. If we look at



(a) The Mller-Lyer Illusion (b) The Kanizsa triangle

Figure 2.5: Example of illusions

figure 2.6 and perceive a vase, we could appreciate the details of its shape, but if we try to look at the shape of the background our perception will shift and we will see two faces and lose the vision of the vase.

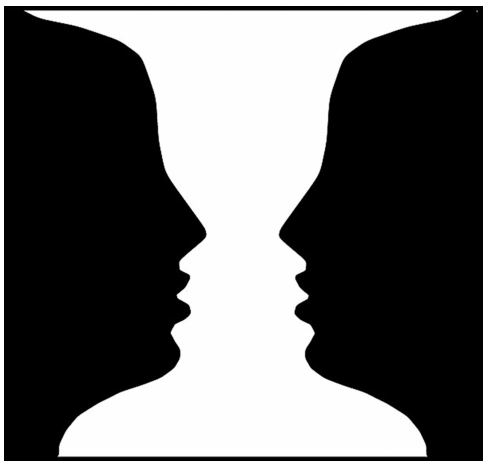


Figure 2.6: The picture devised by E. Rubin in 1915

Ambiguous perceptions are not common, most of the times we are certain of what we "see". We perceive an organised and stable world rather than shifting interpretations of it. In order to explain how we organise the different elements

perceived from a visual scene, the Gestalt perceptual organisation relies on the "law of Prägnanz", also called "law of good form" or the "good Gestalt principle". According to this law dominant percepts are the ones with elements grouped together in a simple, stable, regular... ordered way. The concept was introduced by Wertheimer [1938] and refined with a number of principles or factors:

- Proximity
- Similarity
- Common Fate (trends of motion)
- Closure (filling gaps of figures)
- Direction
- Objective set
- Good curve
- Past Experience

Those factors somehow define what "ordered" elements are. In fact the concept of order is important because it can easily be assimilated to the more abstract concept of "Prägnanz". We have chosen an extract from the introduction of Koffka [1935] to better describe it.

> We speak of an orderly arrangement of objects when every object is in a place which is determined by its relation to all others. Thus the arrangement of objects thrown at random into a lumber room is not orderly, while that of our drawing room furniture is. Similarly we speak of an orderly march of events (Head) when each part event occurs at its particular time, in its particular place, and in its particular way, because all the other part events occur at their particular times, in their particular places, and in their particular ways. An orderly march of events is, e.g., the movement of the piano keys when a practiced player plays a tune; a mere sequence of events without any order

takes place when the keys are pressed down by a dog running over the keyboard (p.15).

In the book Koffka debates if order should be considered as subjective, why should the lumber room be considered more ordered than the drawing room ? If both result from the application of mechanical laws, why should a personal feeling of preference be used to determine whether the room is ordered or not? The Gestalt theory tries to demonstrate that order is a "characteristic of natural events and therefore within the domain of physics" (p.17). The presence of this characteristic could be given by the principles of perceptual organization listed above.

The whole is other than the sum of its parts.

This assertion from Koffka is one of the tenets of the Gestalt theory. In figure 2.5b the analysis of the parts would give us no clue about a white triangle, it is the organization of the parts that we perceive as a triangle. We will not lose the opportunity to use a more complex and beautiful illustration for these ideas. Figure 2.7 shows Monet's painting "La Gare Saint Lazare". Figure 2.8 presents two crops from the previous image. Please note, that image 2.7 has been rescaled to fit in the document, while image 2.8 did not undergo any image rescaling. If we look at the whole image, it is easy to perceive the train, people waiting, the station etc. If we only have a look at the parts of the image where the train, or the people are, perception of the figures becomes harder.

The statement "The whole is other than the sum of its parts" could be taken from a more general perspective. Wertheimer and Koffka use the example of music. One could analyse separately members of an orchestra playing, to eventually discover a formula to predict the note played by each member in a moment of time. However this knowledge would not lead us to explain why each member plays that note at that moment. On the other hand if we listen to all the musicians as one symphony, not only would we know what each musician did but also why he did it, so "the whole performance woud be meaningful" (Koffka [1935] p.18).

Some may find music and symphonies a little bit abstract, so we'll look at something more concrete, a bookcase for example. Let's imagine that we get a

Figure 2.7: La Gare de Saint Lazare - Monet

"do it yourself" model as a surprise present, without instructions. At first we would be puzzled trying to guess which kind of furniture could be build with those boards and screws. An isolated analysis of each piece would hardly tell us the nature of the piece of furniture we need to build. Considering all the pieces we could start filtering options, for example if we do not have table legs, it is not likely to be a table. We could look to each piece in detail and see which kind of screw it needs and derive what pieces may come together. There are a lot of chances that several options come up. An image of the whole bookcase built would serve not only to identify it, but also to explain how to build it. The difference between images of the parts and one image of the whole is that the last one shows the connections between the parts. One single image of the whole carries more relevant information to explain what it is and why those pieces are required, than five images of each piece, "multum non muta".

(a) Train from image 2.7 (b) People from image 2.7

Figure 2.8: Crops from La Gare de Sain Lazare

2.2.5 An empirical theory of vision

Figure 2.9 presents two examples of optical illusions. In our opinion, in the first image the photographer is holding a miniature while in the second one he is not holding anything, just playing with the effect of perspective, but we could hardly argue against someone with a different opinion. In these examples the pattern of light of a small plastic miniature would be the same as the one from a huge steal construction. We could consider these examples as rare cases in which two different sources result in the same retinal output. According to Purves & Lotto [2003] they are not rare, actually every percept may be generated by many sources:

> the sources of any retinal stimulus (and thus its significance for subsequent action) are unknowable directly. Any element of a visual stimulus could have arisen from many - indeed , infinitely many- different objects and conditions. As a result, the output of any detector

(a) Atomium, Real or Fake

(b) Eiffel Tower, Real o Fake

Figure 2.9: Photographies of European Landscape

> to the rest of the visual system is necessarily as ambiguous as the stimulus it presumably encodes (p.5).

Most people fail to find a sheep in figure 2.10, they see snow and grass (please go ahead and try to find it before continuing). However if they could walk further into the field, they would notice that the brown grass is not such, but a mass of hundreds of brown sheep (figure 2.11). In this example there is no artificial trick, just the fact that perception of brown grass could be generated by real grass, by hundreds of sheep, and maybe by many other "objects and conditions".

This "ambiguity" is counterintuitive, Purves & Lotto [2003] recognizes that visual stimuli ambiguity could be "hard to appreciate at first... a quick look around most environments provides a definite and clearly useful sense of the real world" p.5. The authors state that we cannot be sure of the nature of the source of what we "see" (perceive), because what we "see" can be generated by an infinite number of sources. Nevertheless, despite this ambiguity, most of the time we are sure that what we "see" is what it is in reality.

The inherent ambiguity of visual stimuli is "the primary problem with this (Marr's) or any rule-based scheme of vision", according to them the visual system has no means of determining directly the relationship between stimulus and source. Purves & Lotto [2003] claims that what we see is "a probabilistic manifestation of the past rather than a logical analysis of the present" p.11. So, given a visual stimulus, viewers see the "probability distribution of the possible sources of the stimulus" p.10, or in other words a "statistical reflection of visual history"

Figure 2.10: Photography of snow and grass by Liezel Kennedy @pilgrimfarms

(p.227), but not necessarily the "real world". In fact discrepancies between reality and perception are what we refer to as visual illusions.

In the probabilistic concept of vision, a given neuron or group of neurons cannot simply encode image features nor can they encode a particular perceptual quality seen by the observer (p.210). This statement is reinforced by physiological facts, for example LGN inputs come in majority from the cortex and not from the retina. For Purves and Lotto,

> "neural responses are difficult to rationalize in terms of a hierarchical progression from image features detected at the input stages of the visual system to a higher order, convergent representation of those features in neurons whose properties correspond to the perceptions reported by human observers" (p.211)

Instead of that, neuron activity should be considered "in terms of its contribution to the conjoint probability distribution that describes the relative frequency of occurrence of all the possible sources of any component of the stimulus in relation with the rest of the scene" (p.223). The idea is that our visual system is

Figure 2.11: Photography of sheeps by Liezel Kennedy @pilgrimfarms

not encoding and decoding information on which cognitive operations could be based, instead a visual stimulus generates a pattern of neuronal activity that has been formed by the probability distribution of what this stimulus turned out to be in the past. Neuron activity does not encode any feature, it's just a statistical contribution to the probabilistic significance of the pattern as a whole (p.223).

When the authors refer to the "past" they consider both phylogenetic (race inheritance) and ontogenetic (individual experience) contributions. The architecture of the visual system "should be a more or less direct manifestation of the statistical relationships between images and sources experienced over the existence of a species and the lifetimes of its individual members" (p. 219). We could therefore consider a person's perceptual system as unique, with similar base to other members of the species but adapted or tuned with personal experiences.

Another interesting suggestion from Purves & Lotto [2003] is that visual perception and reflex responses are much the same in its overall organization and purpose (p.224). To illustrate this idea, they describe saccades. Saccades are the conjunctive eye movements that direct the gaze of the two eyes to objects of

interest, its frequency is about three times per second. "Saccades usually occur in response to aspects of the retinal stimulus about which the observer has little or no conscious knowledge" (p.223). This means that very often, visual patterns generate neuronal patterns that make our eyes to move without being aware of it. We can of course consciously move them, but it is interesting to notice that consciousness is not required. If we follow this path, visual perception could be defined without awareness.

2.3 Visual attention

In the previous sections we have seen that visual system has to deal with a large quantity of information. Some pretend it is actually "inexhaustible" or "infinite". We have discussed how our perceptual system could represent, describe or organise it. We found answers for questions like "why do things look as they do" (Koffka [1935]) or "how could we construct such a perceptual system" (Marr [1982]). In this section we will review theories about information selection. No matter how powerful human brain turns out to be, if it has to deal with potentially infinite amounts of information, there must be mechanisms to select the most significant one. The system in charge of selectivity is visual attention.

2.3.1 Perception and Attention

We already introduced the concept of attention while describing the theory of information pickup (Gibson [1986]). Perception and attention seem to be closely related. For example, Gibson realized "that perceiving is an act, not a response, an act of attention, not a triggered impression, an achievement, not a reflex." (Gibson [1986] p.149). Gestalt psichologists also treated attention in their works, Koffka [1935] defines attention "as an Ego-object force" that can be either voluntary or involuntary (p.358).

Involuntary attention is called exogenous attention or stimulus-driven capture. Voluntary attention is called endogenous or goal-directed attention. The former corresponds to an automatic orienting response to a location where sudden stimulation has occured. The latter corresponds to our ability to willfully monitor

information at a given location (Carrasco [2011]). They can also be thought of as "bottom-up" or "top-down" processes respectively (Snowden *et al.* [2006] p.259).

Prinzmetal *et al.* [2009] proposes that "voluntary and involuntary attention affect different mechanisms and have different consequences for performance measured in reaction time. Voluntary attention enhances the perceptual representation whereas involuntary attention affects the tendency to respond to stimuli in one location or another". The paper refers to several other differences between the two systems, for example, voluntary attention increases during development whereas involuntary attention decreases with age. Under voluntary attention more perceptual processing resources are allocated which results in a "more veridical perceptual representation" , attended objects are perceptually processed faster and more completely than unattended objects etc. On the other hand involuntary attention "selects the output from perceptual processing". Mack & Rock [1998] exposes fundamental questions about perception and attention:

> What is the relationship between attention and perception ? How much, if anything, of our visual world do we perceive when we are not attending to it ? Are there only some kinds of things we see when we are not attending ?

Despite the general impression that we see nearly everything in our field of view, Mack & Rock [1998] suggest that most of the time we perceive very little if anything of the information catched by our retinas. Attention is often thought of as the mechanism we use to look more closely at some things, but not as something necessary to "see", whereas Mack & Rock [1998] give critical importance to attention in perception, their main hypothesis is that "there is no (conscious) perception without attention" (the word conscious has been added according to further explanation in the book p.13). In their book "Inattentional Blindness" the authors describe a phenomenon, called inattentional blindness, that occurs when healthy people (without vision defects) fail to perceive an unexpected stimulus that is in plain sight.

A popular extension of the "Inattentional Blindness" for dynamic events was presented by Simons & Chabris [1999]. Half of observers failed to perceive a highly salient but unexpected stimulus: a gorilla passing through a group of people play-

ing with a basketball. Observers were told to perform some tasks involving focal attention, for example counting ball passes. The level of inattentional blindness was found to depend on the difficulty of the ordered task. Simons & Chabris [1999] results are consistent with Mack & Rock [1998] findings: "observers fail to report unexpected, suprathreshold objects when they are engaged in another task".

Nevertheless, attention can be captured. Mack & Rock [1998] presents evidences that there are "meaningful stimuli that can attract attention under conditions of inattention and that are thus consciously perceived" (p.18), for example a cartoon-like happy face or seeing his own name. These statements lead to the idea that retinal input from unexpected stimuli are also subjected to extensive processing and only objects to which voluntary attention is directed or the ones that are able to capture attention are perceived. "Attention provides the key that unlocks the door dividing unconscious from conscious perception".

2.3.2 Serial models: FIT and GS

The work Sternberg [1966] introduces the idea that a high speed "exhaustive-scanning" process takes place in memory when subjects judge whether a test symbol is contained in a short sequence of symbols. Exhaustive-scanning means that searching for item i in a list of N items is done serially and requires all items in the list (N) to be classified as targets or distractors before returning a positive (match with target) or negative (no match) answer. Exhaustive scanning opposes self-terminating scanning, where positive answering is returned as soon as a match is achieved. Sternberg [1969] suggests that exhaustive-scanning is used to "determine the presence of an item in the list" while self-terminating scanning is used to "determine the location of an item in the list".

Treisman & Gelade [1980] presents a theory of attention involving serial object evaluation: "A feature integration theory of attention" (FIT). In a first stage features are registered "early, automatically, and in parallel across the visual field", and then in a latter stage requiring focused attention "objects are identified separately". The visual scene is coded along a number of separable dimensions or feature maps. A feature is a particular value on dimension, for example, color

and orientation are dimensions while red and vertical are features on those dimensions. Feature maps are organised retinotopically. According to Treisman and Gelade stimulus locations are processed serially with focal attention in order to recombine separate representations into objects that would be consciously perceived and stored. "Floating-free" features, the ones that have not been recombined, would not be consciously perceived or would "perhaps recombine to form illusory conjuctions" (Treisman [1977]). The fact that unattended areas are not perceived as empty space is explained by means of top-down processing which is "capable of utilizing past experience and contextual information".

Treisman and Gelade clearly differenciate between "feature seach" and "conjuction search", we can "detect and identify separable features in parallel across a display" but "conjuctions, in the other hand, require focal attention to be directed serially to each relevant location; they do not mediate texture segregation, and they cannot be identified without also being spatially localized" (Treisman & Gelade [1980]). Spatial localization is interesting, in the case of features, their identity "can be registered not only without attention but also without any spacial information about their location". Feature localization is considered as a "special kind of conjuction task", a conjunction between feature and spatial location, thus attention is required to perceive correctly the feature and its location. The FIT suggests two ways of becoming aware of unitary objects. The first one is the one we have described in this section, integrating features registered under the same spatio-temporal "spotlight". The second one is through top-down processing. When focused attention is prevented by brief exposure or overloading, the presence of an expected object can be checked by matching its disjunctive features to those in the visual scene without also checking how they are spatially conjoined.

An alternative to the feature integration model for visual search was published by Wolfe *et al.* [1989]. The authors propose that the serial stage described by Treisman & Gelade [1980] could be "guided" by information from the parallel processing. In the FIT if the parallel processes fail to identify a target, the serial stage receives no information other than the registered features. However it would be more convenient if it could guide the next stage. For example if we consider the task of searching for a red X among a group of green Xs and red Os, parallel

processing could differentiate between red and green items. The locations of the green items could be passed to the serial processing stage to avoid wasting time and effort in their analysis. This way information from the parallel processes would guide the serial processing.

Wolfe [1994] presents a second version of its guided search model (GS2) based on "activation maps". An activation map is a weighted sum of feature maps. Like in the FIT, the visual scene is coded along different retinotopical feature maps, but in the case of GS2 feature representation depends on stimulus-driven (bottom-up) and user-driven (top-down) activation component. The bottom-up activation is a measure of how unusual the feature is in its present context. The activation for one location depends on the difference between the value of the feature in this location and the value of the same feature in the neighbouring locations. The author uses a 5x5 matrix to calculate the activation of a particular location, with the location in the central position. The top-down activation is a measure of how important is the feature in the target. If one feature is present in the target and not in the distractors it gets more weight. The activation for one location depends on the difference between the value of the feature and the target value for the feature. "Each feature module can be thought of as a pair of topographic maps with hills of higher activation marking locations receiving substantial bottom-up or top-down activation. Attention is attracted to the hills." The serial process will evaluate the location with more activation, if the target is not identified, attention will shift to the next highest activation location and so forth.

2.3.3 Race models: FIRM and TVA

Bundesen [1987] introduces the concept of race models for selection from multi-element displays. In a race model items are processed in parallel and attention selection is made of those items that finish first (the winners of the race). The selection of targets rather than distractors is based on processing of targets faster than processing of distractors. In Shibuya & Bundesen [1988] the authors propose a fixed-capacity independent race model (FIRM). The model processes in a first stage attentional weights for each item. An attentional weight (w) is a measure of the strength of the sensory evidence that the item is the target. The amount

of processing capacity dedicated to each item is proportional to the attentional weights, so more capacity is allocated to items with higher evidence to be the target. The time required to encode each item follows an exponential distribution with the item's processing capacity as rate parameter.

Bundesen [1990] presents the FIRM as a particular case of a more general theory, the "theory of visual attention" (TVA). TVA integrates into a unified mathematical frame the biased choice model (Luce [1963]) to describe single-stimulus identification (selection of categories) and the choice model for partial report (selection of objects). The previous models were non-process models, but thanks to the race model a process interpretation could be provided.

In the TVA attentional weight of an item (w_x) is determined by summing up products of two factors across all perceptual categories:

$$w_x = \sum_{j \in R} \eta(x, j)\pi_j$$

where R is the set of all perceptual categories, η is the strength of the sensory evidence that element x belongs to category j, and π_j is the pertinence value of category j. A pertinence value is a measure of the current importance of attending to elements that belong to category j.

The rate parameter is determined by the rate equation:

$$v(x, i) = \eta(x, i)\beta_i \frac{w_x}{\sum_{z \in S} w_z}$$

where η is the strength of the sensory evidence that element x belongs to category i and β_i is a perceptual decision bias associated with category i ($0 \leq \beta_i \leq 1$), which represents a measure of the perceiver's general bias toward identifying any presented stimulus as stimulus i. w_x and w_z are attentional weights of elements x and z, respectively. S is the set of all elements in the visual field (definitions from Bundesen & Habekost [2008]).

The weight and rate equations are the two central equations of the TVA. If we combine them v is a function of η (strength of sensory evidence), β (perceptual bias) and π (pertinence) values. We could consider parameter η to be "data driven" or bottom-up, whereas π and β should be considered as top-down

parameters because they are user-driven.

If we suppose η, β, π to be constant during the period in which the stimulus is exposed, the FIRM can be obtained from the TVA (Bundesen & Habekost [2008] p.65).

The author presents TVA as a unified theory of visual reognition and attention selection opposing "early selection" and "late selection" traditional approaches. Early selection theories claim that attention comes before recognition (ex: FIT or GS). Late selection theories claim that pattern recognition is executed before attention (ex: Deutsch & Deutsch [1963], Rumelhart [1970]). According to Bundesen & Habekost [2008], "selection and recognition are neither early nor late in relation to one another but occur simultaneously" (p.43). "In agreement with late selection theories TVA assumes that strength of sensory evidence for perceptual categorizations... are computed before selection takes place... In agreement with early selection theories, the categorical recognition problem is resolved only for those elements that are selected (encoded into the visual short-term memory, VSTM)" (p.44). It is important to note the difference between "holding a representation of sensory evidence and achievement of full recognition", since only in the latter case is a categorical decision about the nature of the object made by the perceptual system.

In Broadbent [1971] selection of inputs is referred to as "filtering" and classification of the selected inputs is referred to as "pigeonholding". In TVA, filtering mechanism is represented by attentional weights. Increasing the pertinence value π of category i in relation with other categories will speed up the encoding process of item belonging to category i. The pigeonholding mechanism is represented by perceptual bias parameters β_i. Increasing the bias associated with category i will increase the v value of categorization that any x element of the visual field belongs to category i.

2.4 Summary

This chapter reviews some of the most important works about human vision published by physiologists and psychologists. Even if visual perception is yet not completely understood, different theories provide valuable ideas that explain

why we see what we do and *what kind of neurophysilogical architecture is able to support the process of perception.* Experiments with humans point different issues relevant for machine perception building.

First, human vision can be wrong. Visual illusions are the main proof that our vision, even in healthy people might be wrong (Coren & Girgus [1978]). In general we have the impression that our vision is right, which can be explained by the fact that very often it is. The probability of information reported by healthy human vision to be correct is in general high. Different theories justify this reliability. Human visual perception is attuned by the fact of perceiving, the more it perceives the more attuned it gets. A human is also capable of improving perception by completing "direct knowledge", "what things told him" with "scientific knowledge", results of reasoning. These are two different forms of learning. Attunement is a direct way of learning, adjusting the system to give better responses. Reasoning over scientific knowledge results in conclusions that might be learned. Both should be considered to improve the probabilities of perceiving.

The theory that perception should be considered as a stochastic process is founded in the fact that visual stimuli are inherently ambiguous. Therefore improving visual perception requires increasing the reliability of the stochastic process behind it. Different factors seem to have influence in reliability, for example the number or quality of features. Showing more features of a particular object increases the response of some determined neurons. Choosing invariant features allows this response to be fired even if the objects change their pose. A third relevant factor is context, some patterns cannot be recognized out of a context.

Second, human vision is selective. We are only aware of a fraction of the objects that we are able to perceive. It is not a problem of recognition ability, it is just that the visual system only reports a part of what could be reported. This behavior can be justified by the amount of potential information that could be perceived, which could require too many resources. Time is a variable that influences perception results, increasing the exposure time might increase the number of objects recognized. This could be explained either by the stochastic or selective character of perception. The selection criteria can be bottom-up, a function of the stimulus, or top-down, a function of the previous knowledge of

the system.

Gibson [1986] challenges some of the most commonly accepted beliefs about vision. According to it neither color nor forms are perceived, but affordances, and such affordances would be directly picked from the structure of the light. These hypotheses give a high-level answer to the fundamental questions about visual perception: *What is visual perception?* and *How is it performed?* Another reference in the field, Marr [1982] proposes different answers. Instead of picking up affordances, vision would be the process by which a description of shapes and positions of things is built from images. Instead of being direct, visual perception would be a sequence of information-processing tasks. Machine visual perception requires its own theory that determines the answers to the fundamental questions.

Chapter 3

Theoretical framework for machine visual perception

The previous chapter describes different answers given by psychologists to the fundamental questions of human vision. These questions are also valid for computer vision. Inspired by the different theories of human vision, in this chapter we propose a new framework to build visual perception systems. We use the three levels of explanation described in Marr [1982] to present our approach. In the first level, "computational theory" we focus on the questions *what is computed ?* and *why should we compute it ?* In the second level level, we describe a formal scheme for representing certain entities or types of information and an algorithm for visual perception. Finally we cover the third level with the analysis of different techniques that could be used to implement visual perception systems.

3.1 Computational theory

Machine perception is usually considered as a pattern recognition problem, however we think that human vision results could be achieved or improved only by considering it as a search problem. Pattern recognition might be a necessary technique to find information leading to the target, but it is not by itself the approach that will give the best results. The computation of visual perception has to integrate time and knowledge, it has to be treated as a process. The goal

of such a computation, finding targets, can be achieved by gathering information. Our theory is inspired by the theory of information pick up from Gibson [1986], by the theory of visual attention from Bundesen & Habekost [2008] and by the etymology of the word perception, "a taking", "collecting" or "gathering" in latin: *"perceptio"*. In order to defend the previous claims, we will analyze the nature of what can be known from an image and discuss why information gathering is more appropriate than pattern recognition to emulate human vision.

3.1.1 What is computed: Categories

What can be perceived from an image? Before defining any process or algorithm we need to understand what is computed in visual perception, the result of the process. We have seen how researchers in human vision suggest different and sometimes opposite answers:

> A description of the shapes and positions of things from images (Marr [1982] p.36).

> Places, attached objects, objects, substances together with events, which are changes of these things. To see these things is to perceive what they afford (Gibson [1986] p.240).

The computation of a shape may be different from the computation of an affordance, which is an unusual concept presented in section 2.2.3. The affordances of the environment are "what it offers the animal, what it provides or furnishes, either for good or ill" (Gibson [1986] p.127). Affordances allow one form to be perceived in several and different ways. For example the image of a house can be perceived as "warm" or "protection", an apple as "eating" or "throwing". For human beings perceiving food, safety or danger is vital, often much more than forms.

The answer we propose for the fundamental question asked in the beginning of the section, is *categories*. It is grounded on the book "Categoriae", where Aristotle intends to classify every object of human apprehension under ten heads:

Substance, Quantity, Quality, Relatives, Somewhere, Sometime, Being in a position, Having a state, Acting and Being acted upon (Reid [1819]). A summary of what Aristotle said about each category can be found in Studtmann [2014].

In the following subsections, we illustrate what different people may perceive from the same image. Then we discuss the nature of categories, which is the first step to understand our strategy to emulate or improve human vision.

3.1.1.1 An Illustration

We start with an example: Velazquez's painting "Las Meninas" 3.1.

Figure 3.1: Las Meninas - Velazquez (Museo del Prado)

The following could be answers to the question "what do you perceive in this image?" given by a person without any particular knowledge of the painting.

- Many people
- A dog

- Adults and children
- Dwarfs
- Men and women
- People from a past period

Most people without interest in painting could answer "people" (Substance), "people from a past period" (Sometime and Substance) or "many people" (Quantity and Substance). Some others willing to describe it, could add "a dog" or "dwarfs". Without specific knowledge in history or in dog breeds it is difficult to give the following answers.

- On the left Velazquez, in the middle Margarita Teresa de Austria. Sourrounding Margarita: Isabel de Velasco and María Agustina Sarmiento. On the right María Bárbola and Nicolasito Pertusato (dwarfs). Behind them Marcela de Ulloa dressed in mourning talking to a bodyguard and at the door José Nieto (identifications by Antonio Palomino).
- A Spanish Mastiff lying on the floor (the dog)
- Maids of honor surrounding Margarita
- A painter at work on the left (Velazquez)
- Court of Felipe IV
- 1656

We can recognize a few extra categories like "somewhere" (on the floor) or "having a state" (dressed in mourning). Next, a list presenting opinions from painters or art experts.

- The true philosophy of the art (Thomas Lawrence)
- Theology of painting (Luca Giordano)
- Representation undertakes to represent itself in all its elements (Foucault [2002])

- Corona Borealis (Lassaigne [1973])
- Velazquez masterpiece "Las Meninas"

Finding the Corona Borealis in the painting requires a big effort of search even for experts in astrology. Stating that it is "Theology of painting" or "The true philosophy of the art" requires a broad knowledge of painting and a deep analysis of the art. The last list is made of answers that could be given by people more or less instructed, not necessarily art experts, who paid some extra attention to the painting.

- Palace master key
- La Orden de Santiago
- Bag of coins
- Dog stepped on
- Large canvas
- More people on the right side

Many people passing by the picture in the museum do not notice that the dog is stepped on. Many more miss the key in Velazquez's waist, and just a few are able to relate it with the position of Velazquez in the king's court or to Velazquez's ambition to present himself as a key figure in the court.

From this example we can draw two conclusions about human vision: first, what is perceived does not depend only on the content of the image but also on the knowledge of the perceiver, and second, the same object can be perceived as several categories. As a consequence depending on the perceiver the same object may have a different set of categories associated. We call *categorization* the process of relating an object with one or more categories.

3.1.1.2 Concept, term and definition

Relating an object and a category is not an arbitrary process. A category is not just a label for a set of objects, it is a concept representing the set of objects. In the work about the acts of mind, Wallace [2011] defines concept as "the internal representation of a thing's essence", which is "both intellectual knowledge, 'that which' (id quod) is understood and the means 'by which' (id quo) the thing known is understood" (p.14). On the other hand a term is only a way to refer to the concept, a sign of it, it is not a definition of it. A sign is "something that shows itself to the senses and other than itself to the mind" (Aurelii Augustini [1]). A term can be arbitrary, different languages use different words (terms) for the same concept, but a definition cannot.

In the picture 3.1 the term "meninas" can be associated to the whole painting, or to some characters of the painting which are different concepts. Terms are sometimes ambigüous but the definition of a concept should not, since it is the means by which the thing is understood. The definition of a concept is an expression of the properties, attributes, qualities or characteristics of the thing represented by it. Since the definition of a concept not only expresses what we understand about the thing but also the characteristics by which we know the thing, it is fundamental for categorization.

> The classical Aristotelian view claims that categories are discrete entities characterized by a set of critical properties which are shared by their members. These properties stipulate the conditions which are both necessary and sufficient to define the intension and extension of a class thus enabling categorization of entities. (Lima & Raghavan [2014])

A category is a concept, whose definition characterizes the members of the category. The definition of a category is the set of critical properties which are shared by its members. We will call characteristics the critical properties of a category. Wallace [2011] states that a definition is not *true* or *false*, it can be good or bad, adequate or inadequate. A good definition is the one that stipulates

[1] *Signum est quod se ipsum sensui et praeter se aliquid animo ostendit; De Dialectica Liber*

the conditions which are both necessary and sufficient to determine whether an object is a member of the category or not. A bad definition may either exclude members of the category, or characterize objects that are not members of the category. In order to emulate or improve the results of human vision we need to be able to build similar or better definitions for categories.

3.1.1.3 Propositional knowledge and knowledge by acquaintance

The definition of a category is knowledge by which the thing known is understood. In the classic philosophy, knowledge was defined as a "justified true belief" (JTB). JTB claims that an agent S knows that a proposition P is true *if and only if* (1) P is true, (2) S believes that P is true and (3) S is justified in believing that P is true. A proposition is a relation between two concepts, called subject and predicate. A predicate can be said, or not said of a subject, a predicate can or cannot be present in the subject (Studtmann [2014]). Unlike concept definitions, propositions always involve truth or falsity. "True" means that what is is, and that what is not is not; and false means just the reverse (Wallace [2011] p.18). We link the idea of justification with the satisfaction of the definition of a category. A predicate, which is a category, can be said of a subject, when the latter satisfies the definition of the former, which is the way of justifying the proposition.

Ichikawa & Steup [2001] suggests that JTB "is an attempt to explicate propositional knowledge, not knowledge by acquaintance". The idea of two kinds of knowledge has already been found in Koffka [1935] (2.2.4). Helmholtz [1995] distinguishes between *"das Kennen"* and *"das Wissen"*. The former is knowledge that consists of "mere familiarity with phenomena" (acquaintance) while the latter is knowledge that consists of knowledge of phenomena "which can be commnunicated by speech" (propositions). Using Helmholtz [1995] words:

> Besides the knowledge which has to do with notions, and is, therefore capable of expression in words, there is another department of our mental operations, which may be described as knowledge of the relations of those impressions on the scenes which are not capable of direct verbal expression. For instance, when we say that we know a man, a road, a fruit, a perfume, we mean that we have seen, or tasted,

> or smelled, these objects. We keep the sensitive impression fast in our memory, and we shall recognize it again when it is repeated, but we cannot describe the impression in words, even to ourselves. And yet it is certain that this kind of knowledge (*Kennen*) may attain the highest possible degree of precision and certainty, and is so far not inferior to any knowledge (*Wissen*) which can be expressed in words; but it is not directly communicable, unless the object in question can be brought actually forward, or the impression it produces can be otherwise represented (p.198).

The knowledge that defines a category may be propositional knowledge, knowledge by acquaintance or a combination of both. Being able to handle both kinds of knowledge might be critical or advantageous for visual perception.

3.1.1.4 Judgment

Judgment is the operation of the intellect by which something is affirmed or denied of something else (Wallace [2011] p.17). A perceptual system is able to affirm or deny that an object o is a member of a category i when it is able to justify the proposition, o is i. This justification is based on the evaluation of the constraints of the category. For machine visual perception, judgment is the computation by which the system evaluates whether an object satisfies the conditions or constraints of the category, the characteristics expressed in its definition.

3.1.2 The computation: Information gathering

3.1.2.1 Knowledge, information and data

In order to compute categories we propose to gather information. Information is closely related to knowledge, the DIKW model (Data Information Knowledge Wisdom) defines it as follows "knowledge is the appropriate collection of information, such that it is intent to be useful" (Ackoff [1989]; Bellinger *et al.* [2004]). The basic idea is to gather information that can be evaluated to determine if it satisfies the constraints of potential categories.

The concept of usefulness is interesting because it conveys the idea of goal. Something is useful when it helps to achieve a goal. The concept collection directly refers to the result of gathering. The word information comes from the latin "formare", to give shape. Therefore information is what has a shape or form. On the other hand the word data comes from the latin "dare", what is given. When a relationship between unstructured data is established, it becomes information. Such information can be related with other information, which is a way of structuring information. Giving signification to a form is an example of relating two types of information, the former is intrinsic to the image, while the latter is extrinsic to it.

We propose that both kinds of information should be gathered to emulate human vision. If we consider visual perception as a mere pattern recognition problem, computation consists in matching information known from a determined category with information found in the image. The main problem is to find the right information. That is why we propose that visual perception should be considered as a search problem, and not only as a pattern recognition one. In fact pattern recognition can be used as an heuristic for the search problem. We will further develop this idea in section 3.2.2.

Visual perception can therefore be seen as a process of gathering information starting from the image. Information are the relations between the different sets of elements of the image. These relations define forms. Information are also the relations between these forms and their signification. Among the different strategies for gathering information, one leading to a useful collection should be chosen. There is a key difference between useful information and the target information. Useful information is not only information directly characterizing the target, but also information that guides the search of the target.

3.1.2.2 Perceptual systems

> The world is specified in the structure of the light that reaches us, but it is entirely up to us to perceive it (Gibson [1986] p.63).

In this section we discuss why a perceptual system should be considered as an intelligent agent and not as a sensor. Our proposal is inspired again by Gibson

[1986], that claims that a perceptual system is "radically different from a sense" (Gibson [1966]), that "perception is not a response to a stimulus but an act of information pickup" (Gibson [1986] p.57).

On the other hand the definition of agent from Russell & Norvig [2014]: "an agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators" (p.35) might be confounding for our purposes, since it associates sensors and perception. Moreover Russell & Norvig [2014] states that "Vision -and all perception- serves for further the agent's goals, not as an end to itself" (p.946). There is no doubt that vision can serve to an agent's goals, but why should not be visual perception a goal by itself, affirming that an image contains (or not) this or the other object ? Why collecting information should not be the goal of an agent, such that we can use the notion of agent for analyzing perceptual systems?

Actually the proposals in Gibson [1986] about perception satisfy Russell & Norvig [2014] definition of agent if we circumvent the differences about sensors and actuators in a visual system. Russell & Norvig [2014] states "A human has eyes, ears, and other organs for sensors and hands, legs, vocal tract, and so on for actuators" (p.35), while Gibson [1986] states that the retina is a receptor that can be stimulated whereas the eye is a perceptual organ, "receptors are stimulated, whereas an organ is activated" (p.53). The eye is one among a hierarchy of organs, including a head that can turn or a body that can move. These organs constitute what Gibson [1966] calls a perceptual system. So Gibson's definition of perceptual system includes not only what Russell & Norvig [2014] calls sensors but also what could be called actuators, head and body. In fact the eye might also be considered as an actuator, Dodge [1903] presents five types of eye movements: fixation, saccadic movement, pursuit movement, convergence and divergence, and compensatory movement. Gibson [1986] completes the list with other adjustments of the visual system: eye blinking, accommodations of the lens, adjustment of the pupil and dark adaptation of the retina (p.216-218). If we consider the structure of light arriving to the retina as the environment, several components of the eye (2.2), as well as the head and body, act upon the environment.

In computer vision the structure of light from a scene is represented in a digital

image. Some devices generating digital images can operate with different resolutions, representing the same captured structure of light with more or less pixels, some can pan, tilt or zoom, others are mounted on moving robots. In computer vision we consider any visual information warehouse or source as the environment. There are many actuators that act upon them. We might think of physical devices like zoom lenses for cameras, but also computer programs that load, crop, rescale, modify or in general process digital images (Gonzalez & Woods [2008]). In this work we will use "visual perception system" or "perceptual system" to refer to an intelligent agent that extracts information from images and whose actions are intended to maximize the probability of finding targets. Hereafter we will only consider actions related with image processing, avoiding anything related with image acquisition. Targets might be a small or large collection of categories of objects. The term object should be understood in a broad sense, an object can be concrete (e.g.: Julius Caesar), abstract (e.g.:number 7), fictional or invented entities (e.g.:beauty, unicorn, honesty etc.) (Nilsson [1998] p.241). Displaying the extracted information on a screen, writing it to a file or sending it through network packets are other kind of possible actuators (Russell & Norvig [2014] p.35) but again not relevant for this work.

We could avoid the discussion about Russell & Norvig [2014] statement: "Vision -and all perception- serves for further the agent's goals, not as an end to itself" (p.946), accepting that the goal of a perceptual system is to fill a database with information from images, but we would be missing a crucial point, the fact that a perceptual system should not be passive like a receptor, it must be active. One of the main ideas we have taken from Gibson [1986] about perceptual systems is:

> such a system is never simply stimulated but instead can go into activity in the presence of stimulus information. (p.53).

In a sensory mechanism, the application of energy stimulus exceeding a threshold can be said to cause a response (p.56). But perception may not depend on the intensity of a stimulus. In section 2.3.1 we presented an experiment where people did not perceive a black gorilla passing through the scene. The fact of missing the gorilla can hardly be associated with a weak stimulation, there must

be another cause. A simple explanation is that they did not expect it, they were looking for other things, or in other words, their goal was different. Our proposal is that the activity of a perceptual system depends on its goals, its rationality [1] and its architecture, which makes the notion of agent interesting for analyzing perceptual systems (Russell & Norvig [2014] Chapter 2).

As mentioned before, the goal of a visual perception system is to find target categories. Each system will have its own targets, which can be more or less specific, for example a cat or an animal. The strategy to find the targets is to pick up as much useful or interesting information as possible. Collected information would be used to evaluate if we have found a target, or if we are closer to finding one. Such computations are what rational agents do.

Rationality at any given moment depends on four things (Russell & Norvig [2014] p.38):

- The performance measure that defines the criterion success
- The agent's prior knowledge of the environment
- The actions that the agent can perform
- The agent's percept sequence to date

In a visual perception system considered as a rational agent, success would be finding the target categories. The performance measure that defines the criterion success would be the probability of finding these targets, the measure of how likely it is that targets are in the image or the measure of how confident the system is about the categorization of a target candidate. The agent's prior knowledge of the environment would be the knowledge about all the information that could be picked up. The agent's percept sequence to date would be the information already gathered.

We resume the computation of a visual perception system using a quote of Russell & Norvig [2014] for rational agents:

> For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given

[1] The meaning of rationality is the one given by Russell & Norvig [2014]

> the evidence provided by the percept sequence and whatever built-in knowledge the agent has (p.38).

A common requirement is to perceive in real time, this means to extract information, such that a human being will have the sensation of immediateness. On the other hand some problems might lead to lapses of minutes or hours to extract information from one image. An agent action requires time and computing power, therefore time and the agent architecture's capacity may limit the possible actions an agent can perform at a given moment. Selecting the right action given the evidence provided by the percept sequence is what we called rationality. The function mapping a percept sequence to an action is called the *agent function* and is implemented by the *agent program* (p.36). Together architecture and program define the agent.

3.2 Formalization of visual perception systems

In the previous section we have defined visual perception as the process by which a perceptual system picks up and categorizes information related to an image. In this section we present how the input of the process, the image, and the output, intrinsic and extrinsic information can be represented. Then we present an algorithm and the primitives to perform the transformation from an image to a set of categories.

3.2.1 Representation

> Representation is a formal system for making explicit certain entities or types of information, together with a specification of how the system does this (Marr [1982] p.20).

> The big problem for AI (Artificial Intelligence) is what to say, not how to say it... the first step in representing knowledge about a world is to conceptualize it in terms of objects, functions and relations... there are many choices about what kind of objects we think might exist in our world. We are free to conceptualize the world in any way we

wish; however some conceptualizations will be more useful (not necessarily more "correct") than others. Next we invent predicate-calculus expressions whose intented meanings involve objects, functions and relations. Finally, we write wffs [well formed formulas] that are satisfied by the world as we have conceptualized it. These wffs will be satisfied by other interpretations as well; we need only to take care that they are not satisfied by other interpretations that our state of knowledge about the world can preclude (Nilsson [1998] p.248).

3.2.1.1 The input: digital images

We have already defined light as one form of energy that is reflected or emitted from objects in the form of electrical and magnetic waves within a particular wavelength range. Different devices have been designed to capture light, cameras and scanners are common examples. The purpose of these optical instruments is to produce a picture or a sequence of pictures to be viewed by people. Details about the different sensors (CCD, CMOS) used for imaging applications as well as other practical uses of digital still cameras can be found in Nakamura [2005]. In this work we will not go any further on how computer images are generated, the technology behind the sensors, but we will focus on what these images are and what can be done with them. We will assume that visual information lies in images and videos (sequences of images) stored in a computer files.

A useful representation of an image in computer vision is a matrix of pixels (picture elements), where the value of each pixel encodes the visual intensity or brightness of the corresponding point in a scene (Gonzalez & Woods [2008] p.55-56). The smallest discernible change in the intensity level is called intensity resolution (p.60). The number of intensity levels in a digital image is based on hardware considerations and is usually an integer power of two, for example 8 bits (2^8) lead to values between 0 and 255, representing black and white respectively. In order to deal with color images, the most popular color space, RGB, encodes the intensities of three components: Red, Green and Blue, using 8 bits for each, 24-bits are used to encode more than 16 Million different colors. RGB is inspired by the three different types of cones in the retina, which are sensitive to different

wavelengths. There are other color models, like CMYK (cyan, magenta, yellow and key (black)) and several color spaces based on the RGB model, like YUV and HSV. For most applications the choice of the model is not relevant, however some works like Podpora *et al.* [2014] suggest that machine vision applications can benefit of a particular model. Gonzalez & Woods [2008] covers different color models and transformations (Chapter 6).

The size of a digital image in a computer, the number of pixels, is usually referred to as pixel resolution or just resolution and expressed with the set of two integer number, the first being the number of columns in the matrix (width) of pixels and the second the number of rows (height). When talking about image quality it would be more accurate to talk about spatial resolution, which quantitatively is the number of pixels per unit distance (Gonzalez & Woods [2008] p.60). If we assume the same scene is represented, then a higher pixel resolution is equivalent to a higher spatial resolution. Figure 3.2 presents three images of the same scene with different resolutions. In the image with high resolution we can read the numbers of the car's license plate "9557", whereas in the other two images this data is lost. High resolution images allow us to perceive more details from the image, another example are the wiper arms. In the low resolution image it is impossible to be sure if the car has or not wiper arms, the distance represented by each pixel is bigger than the size of the required details.

Figure 3.2: Resolution Test (Image from Wikimedia Commons)

Resolving power is the ability of an optical device to distinguish two adjacent points which are close together into individual images. Resolving power depends on spatial resolution and intensity resolution, since two points with the same

intensity will be indistinguishable. The resolving power is defined as the reciprocal of the limit of resolution (either a distance or an angle) (Born & Wolf [1999] p.461). In the image 3.2 we can play with the resolution power of our eyes by zooming in the different images (or moving away and getting closer if you are not reading the digital document), pixels will appear and disappear. If the reader steps back enough (the distance depends on the size of displayed image) the three images are perceived to have the same quality. In the 15-inch screen used to write this document, we have zoomed in the three images of 3.2 separately, so that each of them covers the whole screen. Recognizing a car in the left sample is not evident, specially for people that have not watched it before and have no prior idea about it. Intuitively, being able to distinguish the small details of a picture makes the whole harder to recognize.

For digital images spatial resolution limits the amount of potential information, by definition, nothing will appear when zooming in a pixel. For example, if we have one pixel per meter, we will not be able to distinguish details smaller than one meter. Some image-forming devices can generate images with different resolutions, others can pan, tilt or zoom, making details easier to see. Computer programs can also perform several operations over an image to shrink or zoom it (Gonzalez & Woods [2008] p.65). The latter, called digital zoom, may be less accurate than the former because they are based on estimations (Nakamura [2005] p.243).

3.2.1.2 Intrinsic information

> The visual world can be regarded as being composed of smooth surfaces having reflectance functions whose spatial structure may be elaborated (Marr [1982] p.44).

> The medium is separated from the substances of the environment by surfaces (Gibson [1986] p.22).

In digital images surfaces are sampled and quantized in sets of pixels, whose separation is not always obvious. The intrinsic information of an image are the spatial structures of the set or subsets of pixels composing it. Structures, space

and change are some of the topics covered by mathematics. In this section we discuss how different branches of mathematics may contribute to transform data into information.

Functions and relations. A digital image can be represented by a function $f(x, y)$ mapping the coordinates (x, y) to the value of the pixel (Gonzalez & Woods [2008] p.55). To illustrate the following ideas in a more comfortable way, we will use examples with just one row.

In mathematics a function is a particular case of a relation where each input is mapped to exactly one output. Formally speaking a relation is a set of tuples with objects related, in the case of a function, each element of the input appears in just one tuple, whereas in a relation, it can appear in several tuples. For example the set of pairs (2-tuples) $\{(1, 1), (2, 0), (2, 1), (3, 0), (4, 0), (4, 1)\}$ describes a binary relation R_1 between two sets, the first one being the indices of a row with 4 elements $\{1, 2, 3, 4\}$ and the second one the set of possible values or outputs $0, 1$. Indices 2 and 4 appear in two pairs each, thus R_1 is not a function. A binary relation between two sets A and B is defined by a subset of pairs of the Cartesian product of the two sets.

The application of the relation R_1 over the set $\{1, 2, 3, 4\}$: $\{R_1(1), R_1(2), R_1(3), R_1(4)\}$ can have different results: {[1001] [1101] [1100] [1000]}. While a function can represent an image, a relation can represent a set of images. In section 3.1.1 we claimed that a category is a set of objects that satisfy the definition of a concept. We could say that the set of all the possible results of the application of a relation over a set of coordinates is a category and the relation between *all* the indices and values is a property of the category.

Analytic geometry. Descartes [1897] introduced the idea of using a coordinate system to study Geometry, such that geometrical shapes could be defined and represented with functions, equations or vectors. This branch of Geometry is called Analytic Geometry and leverages several techniques from Linear Algebra. Linear Algebra studies objects with a structure of vector space (Fraleigh *et al.* [1995]), for example matrices. Since images might be represented by matrices, Linear Algebra techniques like linear transformations, linear equations or linear

least squares resolution, can be used to extract properties, like determinants or eigenvalues.

Differential geometry. Analytic Geometry is continued by Differential Geometry. The foundational "Theorema Egregium" proved by Gauss [2007] claims that "a Gaussian curvature can be expressed solely in terms of the first fundamental form coefficients and is therefore an intrinsic property" (Bronstein *et al.* [2008] p.35). Informally, a Gaussian curvature can be expressed using angles, distances and their rates on the surface itself, without references to the particular way the surface is deployed in the Euclidean Space (leaving aside the coordinate system). The properties of surfaces preserved through deformation, twisting or stretching are studied by Topology. A surface can be represented by a set of its invariant properties.

Digital geometry. Digital images result from a process of digitization. The branch of Geometry dealing specifically with the study of geometric or topological properties of sets of pixels is Digital Geometry. Klette & Rosenfeld [2004] presents Digital Geometry as well as related disciplines: Affine Geometry, Projective Geometry, Vectors and Geometric Algebra, Graph Theory, Topology, Approximation and Estimation, Combinatorial Geometry, Computational Geometry, Integral Geometry and Mathematical Morphology.

Measures.

> A function that takes pictures into numbers is called a picture property; a function that takes k-tuples (e.g., pairs) of pictures into numbers is called a relation among (or between) pictures. This chapter defines classes of picture properties, such as predicates, local properties, linear properties, and invariant properties. Particular attention is given to the study of moments, which are an important class of linear properties (Klette & Rosenfeld [2004]).

In mathematical analysis functions that assign numbers to sets or subsets are called measures. Moments, for example are a specific quantitative measure of

the shape of a set of points used in statistic. In general, statistic is the measure of some attribute of a set of data. "Generally speaking in the problem of shape similarity we are looking for a quantitative measure of distance between two shapes" (Bronstein *et al.* [2008] p.3). So, besides of mapping coordinates to values, functions can also be applied to map images (pictures) with values. This kind of functions represent properties of the image, whose expressions belong to the concept's definition. The set of images in a category satisfy the concept's definition, which is a common set of properties represented by a set of functions.

Relations between parts. Instead of mapping a whole set of pixels, we might map only a subset of pixels to a number. Measures can be assigned to parts or regions of an image, just like to the whole. Since they have different elements, the properties of the whole and the parts may be different. In the last quote the authors refer to functions that map a k-tuple of images into a value as "relations". Each part of an image is an image, therefore we might also get properties of the relation between the parts of an image. A particular case of relations between parts are derivatives. A derivative of a function $y = f(x)$ is a measure of the rate at which a value y (e.g. the value of a pixel) changes with respect to the change of a variable x (e.g. coordinate).

Relation of the whole and its parts. The relation between the parts and the whole is the relation of inclusion. Gibson [1986] suggests that "inclusion" is the relation that defines "locus": "the optic array" should be conceived as a nested hierarchy of solid angles and that the optic "array is more like a hierarchy than like a matrix... in an ambient hierarchical structure, loci are not defined by pairs of coordinates for the relation of location is not given by degrees of azimuth and elevation (for example) but by the relation of inclusion" (p.68). A whole can be represented by a subset of its parts, we can just associate the whole to a concept whose definition expresses the relation of its parts. The properties of each part are "propius" to the part (the part's own) considered as a whole itself and therefore does not have to be properties of the whole. The same concept cannot be associated to the whole and one of its parts (or a subset) unless both have the same set of properties, in other words, unless both are members of the

same category.

Segmentation. To get a rough idea of the amount of parts or regions a digital image can be divided into, we simplify the problem assuming that a part must be rectangular, so that we can use the formula to find the number of submatrices of a matrix ($\frac{m(m+1)n(n+1)}{4}$). For example considering the resolution of the picture 3.1 (1125x675) we get thousands of millions of regions. If we add non rectangular regions the number would be even larger. The process of dividing an image into regions is called segmentation and is one of the most difficult tasks in image processing (Gonzalez & Woods [2008] p.689).

Feature extraction. If the amount of possible regions is massive, the number of possible properties is even larger. Different functions can be applied to the elements of each part to extract properties, several relations can be established between the parts and the whole, and even more can be established between parts. All these relationships are properties of the image, and their number multiplies the number of parts. When trillions of features might be extracted from a single image, the challenge is to find the ones that are useful for the goal of the computation.

Numeric features are the ones whose values are numbers. Characteristic functions are functions mapping features with categories, and thus the way to represent the constraints that determine the membership of a category. Categorical features are the ones whose values are represented by a term. Characteristic functions can map both numeric and categorical features into categories. Characteristic functions are a form of proposition where what is evaluated is the subject and what is characterized is the predicate.

3.2.1.3 Extrinsic information

Now, let's suppose that we want to build a spanish people detector in pictures. The detector could carefully extract all the intrinsic information from picture 3.1 trying to find something that characterizes a "spanish". Another option is to use the known fact that Velazquez was spanish. This knowledge is not in the image, it is something extrinsic.

Visual perception systems can be seen as agents that give signification (meaning) to images. Nilsson [1998] suggests that in Artificial Intelligence "Semantics has to do with associating elements of a logical language with elements of a domain of discourse. Such associations are what we refer to as 'meaning'" (p.222). Intrinsic information has to deal with geometric and topological properties, about how the image is organized. However other kinds of information can be associated to a structured set of pixels, information that does not say anything about the spatial structure of an object. This is what we call extrinsic information of an image. What can be said of something is the predicate of a proposition and this something is the subject. Extrinsic information can be expressed with propositions.

Relation of propositions. Let's consider the concepts *enemy, danger, weapons and target*. When we detect an *enemy* we can affirm "there is a target". However if we detect an *enemy* and *weapons* we can affirm "there is a danger". The relation between *enemy, weapons and danger* does not say anything about the spatial structure of the objects or the scene, but in order to categorize the *enemy* as *target* or *danger* we need the information "if there is an enemy and weapons then there is danger" and "if there is an enemy and no weapons then there is a target". "There is an enemy", "there is danger" or "there is a target" are propositions about the world. A perceptual system can categorize a set of pixels as *enemy* and express "this set of pixels is an enemy", a proposition where "the set of pixels" is the subject and *enemy* is the predicate. However to affirm "this set of pixels is a danger" requires a relation between two propositions "this set of pixels is an enemy" and "this set of pixels is a weapon" and the conclusion "this set of pixels is a danger".

Propositional calculus. Propositional calculus provides a language to represent propositions and the relations between them. The elements of the language are atoms and connectives. Atoms are True, False and any string of our choice that will be associated with a proposition about the world. The connectives are $\vee$ (or) $\wedge$ (and) $\supset$ (implies) $\neg$ (not). A sentence or well-formed formula (wff) is either an atom or atoms connected by connectives. The language has rules of

inference which can produce additional sentences. For propositional calculus, we associate atoms with propositions about the world (Nilsson [1998] Chapter 13). The previous example could be represented by E (there is an Enemy), A (there is a Weapon) , D (there is Danger) and Ta (there is a Target) and the expression $E \wedge A \supset D$ and $E \wedge \neg A \supset Ta$.

Predicate calculus. Propositional calculus let us represent propositions and express relations between them. However with propositional calculus we cannot talk about the objects, we cannot express properties about them and hence we cannot represent the definition of a category. Predicate calculus has objects, also called individuals, functions on these individuals that map n tuples of individuals with individuals (number 10 and 2 mapped into quotient 5) and relations over individuals (*Loaded(w), Bigger(4,2), Brother(John, Bill)*), also called predicates (Nilsson [1998] p.241). We can create sentences using connectives (like in propositional calculus) and also quantifiers that let us express properties of entire collections of objects instead of enumerating the objects by name. More details about predicate calculus can be found in Nilsson [1998] (p.239-268) and Russell & Norvig [2014] (p.290-320) (predicate calculus is also called first-order logic). Predicate calculus has all the elements to represent objects, categories, properties and knowledge.

Using predicate calculus we can associate the string "V" with the object "Velazquez", the predicate *WearsS(x)* with the function that maps a person x with True if he wears "the symbol of la Orden de Santiago" and False if he does not, and the predicate *MemberO(x)* with the function that maps a person x with True if he is a member of "la Orden de Santiago" and False if he is not. Thanks to this interpretation we can express: "Velazquez wears the symbol of la Orden de Santiago": $WearsS(V)$ and infer using the propositional sentence $(\forall x)[WearsS(x) \supset MemberO(x)]$ that Velazquez is a member of la Orden de Santiago: *MemberO(V)*. Predicate calculus gives us a simple way to express predicates that represent characteristic functions: *MemberO(x)*, which indicate the membership to a category "Orden de Santiago", and also properties of the category's members: *WearsS(x)*.

Predicate calculus does no more than provide a uniform language

in which knowledge about the world can be expressed and reasoned about (Nilsson [1998] p.248).

3.2.1.4 Knowledge base

In AI knowledge about the world is represented by a collection of sentences called knowledge base (KB). Ontology is the study of the relationships between categories, which "organizes everything in the world into a hierarchy of categories" (Russell & Norvig [2014] p.444). Intuitively organization is fundamental for any search problem. When something is organized search requires less resources. Koffka [1935] and other works from the Gestlalt (2.2.4) discuss the concept of order and propose laws to group elements together. Our definition for organization is: *a set of relations that simplifies search.* Therefore the better the ontology of a perceptual system is the less resources the system will require to find targets.

Gruber [1995] defines ontology as "an explicit specification of a conceptualization". Conceptualization usually starts with the acquisition of knowledge from another entity. Expert systems for example implement a conceptualization given by a human expert in a field of application. Formal languages are therefore the first choice to create conceptualizations. One of the advantages of formal languages is that we have well studied mechanisms to reason about well formed formulas (wffs), and therefore automatically create new propositions, which are knowledge.

Perception is the other way, by which information can be acquired. However the information in an image, its intrinsic information (3.2.1.2) is not easily expressed in words, or in general with formal languages. This difficulty has sometimes been understood as the impossibility to express the knowledge by which an object can be recognized:

> What is the real shape of a cloud?... or of a cat? Does its real shape change whenever it moves? If not, in what posture is its real shape on display? Furthermore, is its real shape such as to be fairly smooth outlines, or must it be finely enough serrated to take account of each hair? It is pretty obvious that there is not answer to these questions - no rules according to which, no procedure by which, answers are to

be determined (Austin & Warnock [1964] p.67).

We think that once again, only propositional knowledge has been considered. Answers to the previous questions might be given by knowledge by acquaintance, which is easier to represent using relations of geometric properties. Geometric properties can be represented using mathematics [1], which can be seen as a formal language, or by a collection of images. Storing a set of images of a category may not be the most efficient way of representing intrinsic information, but is one possible way.

Knowing is an extension of perceiving (Gibson [1986] p.258).

3.2.1.5 Uncertainity

We think that Austin & Warnock [1964] is also missing another critical point: the stochastic nature of perception. In the first chapter we have seen how our confidence in human vision might be misleading. The number and complexity of the structures in the visual world, challenge the constraints that characterize categories. Indeed, it is not easy to define "the real shape" of a cat, just like many other objects. Organic beings grow older, non organic get deteriorated or renovated. Categories are not closed, new objects come out everyday. Compare today's phone and the ones from our old parents. The number of potential layouts under which objects are perceived is large and each layout may influence in its geometric description (different angles, levels of occlusion, positions etc.).

Logical approaches like entailment, theorem proving or propositional model check (Russell & Norvig [2014] Section 7) , are unfeasible or impossible in real world problems. Either the number of models is too large to be computed or unknown. When we cannot create absolute constraints to define categories, there is a degree of uncertainty about the categorization that must be represented.

The main tool for dealing with degrees of belief is probability theory. The ontological commitments of logic and probability theory are the same - that the world is composed of facts that do or do not hold in any

[1] in ancient greek mathematics means "that which is apprehended"

> particular case - but the epistemological commitments are different: a logical agent believes each sentence to be true or false or has no opinion, whereas a probabilistic agent may have a numerical degree of belief between 0 and 1 (Russell & Norvig [2014] p.490).

Unconditional or prior probabilities refer to degrees of belief in propositions in absence of any other information (Russell & Norvig [2014] p.493), so when the number of different objects increases the prior probability that one of them belongs to a category decreases. Conditional or posterior probabilities refers to the degree of belief in a proposition given another proposition. Given proposition A and B we write $p(A|B) = \frac{p(A \wedge B)}{p(B)}$ to express the probability of A given B. $p(B)$ is the probability that an object satisfies B, $p(A)$ is the probability that an object belongs to category A. We wish $p(A \wedge B)$ to be as close as possible to $p(B)$, so that few objects satisfy B and do not belongs to A and few objects that belong to A do not satisfy B. Such a constraint would be a perfect characteristic of the category. Since the cardinality of some categories may be unknown, we will need to estimate probabilities, and hence the importance of a stochastic approach.

We illustrate how intrinsic and extrinsic information can be combined in a stochastic process to find a target of which we have no description. Let's imagine that we want to recognize Margarita Teresa de Austria but we have no description of her. However we do have features that let us recognize from an image the following categories Felipe IV, Mariana of Austria, a face, a child, an adult, a girl and a boy. If we also have extrinsic information, for example the fact that only one daughter of Felipe IV and Mariana of Austria grew older than 1, we can deduce that Margarita is one of the three girls in the middle of the painting. The process would be, we recognize Felipe IV and Mariana de Austria. There is a high probability that people in the picture are related to them. We can recognize the face of three girls, and infer that one of them is Margarita. Information about Felipe IV could be avoided if information about Mariana of Austria is included. However information about Mariana of Austria cannot be avoided since Felipe IV had another daughter with Isabel of Borbón.

Another option would be to select information about las Meninas. If the system can recognize the painting, faces, children, boys and girls and knows that Margarita is one of the main characters of the painting we could deduce that the

best represented face among the three girls corresponds to Margarita's. Each way requires different knowledge and offers different levels of certainty. Joining the probabilities of both may give a stronger evidence. Once we have deduced the identity of the girl in the middle of the image we can grasp the features from the spatial structure of Margarita's face and add it to the knowledge base, so that next time she can be recognized directly, for example in an isolated crop of the painting.

In the first case it is the spatial structure of different parts of the painting related with extrinsic information about the target that leads us to the result. In the second case it is the spatial structure of the whole related with extrinsic information about it that ends up in the same conclusion.

3.2.2 The process of visual perception

In the first section of this chapter we claim that visual perception systems able to emulate human vision cannot be modeled as sensors, a more powerful model is required. The concept of rational agent has many advantages to implement our theory. In this section we show how a perceptual system can be modeled as an intelligent agent. We present its possible actions and an agent program to efficiently gather information from images. Finally we discuss how visual perception systems can be improved through learning.

3.2.2.1 Processing modules

In section 3.1.2.2 we discuss why perceptual systems should be considered as intelligent agents and not as sensors. The first step in designing an agent is to specify the task environment, which is done by describing the P.E.A.S.: Performance measure, Environment, Actuators and Sensors (Russell & Norvig [2014] p.41).

We have limited the environment to information from digital images. These are a sampling of the world created by image acquisition devices. The goal of the agent is to find target categories represented in the images. Sensors are the components of the agent that evaluate data, sets of pixels, from the image to categorize it. A sensor has built-in the definition of a category and the constraints

that characterize it. It evaluates in what degree a set of pixels satisfies these constraints and therefore gives a degree of evidence that this set of pixels represents an object member of the category. Sensors implement the transformation of an image into a set of intrinsic information. In this paragraph we have used the term sensor to comply with the acronym P.E.A.S., which is widely used in AI. However for visual perception systems we prefer the term *recognizer*, the term sensor would be better used for cameras. The set of *recognizers* of a perceptual system is called the *recognition module*.

Images can be a very large collection of data but objects may be represented only by a part of it. When a sensor evaluates the wrong set of data it will not be able to extract the right information. We consider that segmentation methods are actuators that divide an image into parts, so that *recognizers* can evaluate them. *Segmenters* implement the transformation of an image into a set of images. The set of segmenters of a perceptual system is called the *segmentation module*. Perceptual systems may have other kinds of actuators, that we group in the *pre-processing module*. Pre-processing techniques seek to improve the conditions of the image and are usually filters, for example noise filters. These techniques transform an image into a different image. Segmentation is often seen as a pre-processing method, but we think that segmentation is much more. The results of segmentation by themselves are characteristic. Segmentation can be seen as a function mapping an image with a number of segments. Moreover, segmentation methods evaluate data and categorize pixels according to the degree of satisfaction of a set of constraints, just like *recognizers*. Sometimes, the boundary between *segmenters* and *recognizers* fades out, when recognition is performed with the features given by the segmenter, or when segmentation is based on pattern recognition.

The performance measure is the evidence that the target categories are represented in the image. The evidence given by the recognizer of each target category can be used to calculate a performance measure, but they are only one part of the parameters. Let's take for example a scene categorized as "beach" and another as "city". If the target is a "palm tree", the probability of finding one in an image of a beach is higher than in the city. These evidences are not given by recognizers, we need another kind of component that we call *rational module*.

The rational module manages the KB, the component that *reasons* about the intrinsic information extracted from the image and the extrinsic information represented in the KB. The conclusions of this reasoning provide *rational evidences* that are computed together with the *geometric evidences* given by the rest of the modules to categorize an image. The rational module also has a second function, to guide the process of search. It is the program of a rational agent mapping the information gathered with actions, the component that decides the next action as a function of the collected information.

3.2.2.2 State space

State space is a common representation for search problems in AI (Nilsson [1998] Part II). In our Thesis for visual perception a state must describe the information that has been collected. A state has a list with all the categories from which the system has information. This information can be a geometric degree of evidence η given by the recognition module, a rational degree of evidence β given by the rational module or a degree of pertinence π[1] associated. The degree of pertinence represents the importance that has been given to this category by an external source, not by the visual perception system. A target category is very pertinent for example, but may not be represented in the image.

Hierarchical structure. A state has also a list with segments of the image. Each segment is itself an image, from which information can be collected. We implicitly associate a segment with the set of pixels it represents, but do not express it explicitly. Each segment has its own list with all the categories and one with its own segments. The list of categories should include all the known categories, but we can assume that any known category not present in the list has η, β and π equal to zero. Finally for each segment we can include a reference to show if a category has been activated for that segment. We use "category activation" to represent the output of a visual perception system at a given moment. Even if the same segment might be associated with several categories, this cannot happen at the same time. A screwdriver can be perceived as a tool or as a weapon, but

[1] We use a nomenclature inspired by Bundesen & Habekost [2008], where the geometric evidence is called sensory evidence and the rational evidence is called perceptual bias

not both at the same time. The data structures used to represent states can be described as follows (3.2.2.2).

$$categoryList \equiv \Big((category_1, \eta_1, \beta_1, \pi_1), ...(category_n, \eta_n, \beta_n, \pi_n)\Big)$$
$$segmentList \equiv (segment_1, ...segment_n)$$
$$segment \equiv [activationId, categoryList, segmentList]$$

The name of each category is arbitrary, it is just a term, a way to refer to it. The information about the categories could also be grouped in three lists, one for the values of η, another for the values of β and another for the values of π but in this case we would need to explicitly include all the categories with value 0. These structures represent information in a hierarchy, in which the whole image is at the top and is divided recursively into smaller regions.

$$State \equiv [activationId, categoryList, (Image)]$$
$$Image \equiv [activationId, categoryList, (region_1...region_n)$$
$$region_n \equiv [activationId, categoryList, (subregion_1...subregion_n)]$$

When the state space graph becomes too complex, an implicit representation can be given by three components: (1) a description of the start node, (2) functions to transform a state description representing one state of the environment into one that represents the state resulting after an action, these functions are usually called operators, and (3) a goal condition (Nilsson [1998] p.130).

Start state. The start state has no category activated and a category list made of categories judged pertinent (with π value different of zero) or with rational evidence drawn from previous images, for example in videos. The segment list contains one segment, the whole image, which is a copy of the start state but with the segment list empty. In specific search, targets are pertinent, but also categories related to the targets, for example if the target is "city", the category "building" might also be relevant.

Goal states. The goal states are the ones, where *sufficient* evidences have been found for a *sufficient* amount of target categories for a determined image. The specification of *sufficient* is different for each problem. We can group them in two classes, "specific search" and "description". In specific search problems the targets are specified a priori: "search for cats". In description problems anything can be a target: "tell me what is represented in the image". In the first type of problem, the number of goal states is usually small, whereas in the second it is very large.

Transitions. The transition from one state to another is the effect of the actions segmentation, recognition and reasoning, which in this context are called operators. The operator *Recognize* provides geometric evidence η_i from an image I for any category i which has a recognizer in the recognition module. The operator *Segment* provides a set of regions S generated from image I. The set of regions is represented in the state as the segment list of image I. Each segment has its own category list and segment list. The first one is initialized as the start state, except for the β_i which are computed using information from the segment's parent and an empty segment list.

We use the term region to refer to a segment of an image. There is no difference in the nature of an image (I) and a region (R), I is just the first R. The rational module has three processes: *Categorize*, *Divide* and *Comprehend*. The first one acquires information with the operator *Recognize* and activates a category for the region. The second one is based on the operator *Segment*. The process of division can be iterated to build a hierarchy of segments. The decision of dividing into more parts is computed with the information gathered by the processes *Categorize* and *Comprehend*. The latter acquires information with the operator *Reason* and also activates a category for the region. These processes define the agent's program.

The number of transitions depends on the complexity of the image, the knowledge of the system and its goal. A complex image is likely to have more objects and therefore require more segmentation. More knowledge means a bigger KB that could be explored, and thus more propositions might pop up. The effect of these factors is an increase of the number of transitions.

3.2.2.3 Processing strategy

Exploring human knowledge can be extremely expensive. The dictionary of "La Real Academia Española de la Lengua" has more than 90.000 entries (www.rae.es), Diller [1978] assumes that Webster's dictionary contains more than 450.000 entries. The number of possible relations between these entities is unapproachable. In fact it is likely that for any category a relationship can be found with any other category. This means that when a region is categorized, the rational module could eventually pop up every single category known by the system.

Knowledge pruning. To avoid this, we can limit or inhibit much of the knowledge that will be used. For example we could use a domain specific ontology or a taxonomy. A taxonomy is usually a hierarchy of concepts defining relations of subcategory or supercategory, whereas an ontology studies any type of relation between categories. Another advantage of using a taxonomy is that we can avoid recognizers for categories like "animal" or "food". It seems difficult to find a geometric property shared by a lion, a snake and a cow. However a domain specific ontology may be necessary when specific search deals with abstract categories. For example, "window" is not likely a subclass of "danger", however a relationship between them could be found in an ontology about house risks.

Heuristic search. Exploring an image can also be expensive. In the megapixel image representing "las Meninas" (3.1) we could easily extract millions of different segments (3.2.1.2). For many real world problems, an exhaustive search of the space generated by the relations between categories and segments could be unfeasible or expensive. In such cases, AI problems can be approached with informed or heuristic search. We take advantage of the agent's knowledge, either from its KB or from its percept list, categories gathered, to guide the search.

The goal states are defined by a minimum of evidence that has to be reached on a minimum of categories. A natural heuristic is the difference between the evidence of the candidate state and the target states. The closer the better. The information gathered from the image provides new categories but also allows to update the rational evidences β of related categories. Heuristics are used within an evaluation function that determines whether the perceptual system should

segment a region, analyze another segment, or integrate the information. For example, given an image segmented in three regions, we have categorized one of them as a car and another as a bicycle. If we are looking for a wheel, should we segment the region with the car, categorize the third region or integrate the categories "car" and "bicycle"? The agent has to select which process will be executed next: *Categorize* , *Divide* or *Comprehend*.

After each segmentation the agent has to select which region should be categorized first. We suggest a method based on the rate parameter equation from Bundesen & Habekost [2008].

$$v(x,i) = \eta(x,i)\beta(x,i)\frac{w_x}{\sum_{z\in S} w_z} \qquad \text{(Eq.1)}$$

where S is the set of segments, i is a category and

$$w_x = \sum\nolimits_{j\in V} \eta(x,j)\pi_j$$

where V is the set of all the categories that can be recognized by the recognition module. Instead of a fixed and global β_i, its value undergoes different changes by the process of perception in function of the segment.

Recognize transforms pixels into geometric degrees of evidence, each recognizer i, gives a η_i value. *Categorize* transforms a set of segments into a pair composed of a segment with activated category, it affirms that a region is one category or the other.

The affirmation that a region is a category is done implicitly by selecting the region x with the highest $v(x,i)$, which at the same time is relating x with the category i. The process of visual perception is not about reporting meaningless terms, it is about attuning a system. Maybe now the quotes from Gibson [1986] and Purves & Lotto [2003] make more sense in a context of machine visual perception:

> such a system is never simply stimulated but instead can go into activity in the presence of stimulus information (Gibson [1986] p.53).

> the visual system is not organized to generate a veridical representation of the physical world, but rather is a statistical reflection of

visual history (Purves & Lotto [2003] p.227).

When different regions have been categorized the system can integrate them into a whole. It is the process that we have called *Comprehension*. Segments can be combined in different ways, and thus another selection must be done. We use a formula inspired by the previous one, but instead of several segments, we have several combinations of categories as input.

$$u(x,i) = \beta(x,i)\frac{w_x}{\sum_{z\in P} w_z} \qquad \text{(Eq.2)}$$

where P is the set of possible combinations made with the active categories from a region's segment list, i is a category and

$$w_x = \sum\nolimits_{j\in Z} \beta(x,j)\pi_j$$

where Z is the set of all the categories known by the system and $\beta(x,j)$ is the rational evidence that a combination of categories x can be related with category j.

The main difference between *Comprehend* and *Categorize* is that the former updates the rational evidence for some categories of the segment (β) and the latter updates the geometric evidence (η).

A state can be evaluated at any moment to check if we have reached a goal state. We suggest that goal states should be defined by evidences for categories and not by active categories. The set of active categories could be assimilated to momentary perceptions, while a state represents the collected information. States are the results of perception over the time, more time means more information gathered. Over time, an image or one of its region can have different categories activated. Given a list of target categories, a state should have sufficient information for the perceptual system to find most of the categories that have been activated in previous states. We could say that states have memory, maybe representing something like the Visual Short Term Memory (VSTM) for humans.

3.2.2.4 Processing algorithm

In this section we propose an algorithm for the processing strategy presented in the previous section (3.2.2.3). We have divided it in four processes: *Visual Perception*, which is the main process, *Categorize*, *Divide* and *Comprehend*. When a variable appears both as a result and a parameter of a function, its content has been modified. As stated before, the goal of the computation is to gather information, which is collected in the variable *state*. The functions, actions, or operators that add the information are:

- *Recognize*: this operator modifies the values η in the *categoryList* of an image or region.
- *Segment*: this operator directly modifies the *segmentList* of an image or region, but also modifies indirectly the values β with a call to function *Comprehend*.
- *Reason*: this operator modifies the values β in the *categoryList* of an image or region.

We have two selection mechanisms, *isBetterThan* and *SelectAction*. The former one selects a pair of category and region to be categorized or a pair of category and set of information to be categorized. The latter selects the process with more evidence to lead to a goal state.

The variable *state* is also modified by two other functions: *Activate* and *Initialize*. The first one sets *activationId* to the last category found for the image or region, while the second one is used to create the data structures in the beginning. We will further discuss the role of these functions in the next section (3.2.2.5).

Process 1 Visual perception

Require: $\mathcal{I}$, an image
Require: *pertinentList*, a list with pertinent categories
Require: *targetList*, a list with target categories
Require: *machine*, a machine with *previousState* and non empty recognition, segmentation & rational modules
Require: *state* , state with *activationId*, *categoryList* & *segmentList*
Require: *segmentList* empty

```
state ← Initialize(state, previousState, pertinentList)
scene ← Initialize(scene, previousState, pertinentList, I)
push scene into segmentList
repeat
    scene ← Categorize(segmentList)
    nextAction ← SelectAction(machine, scene, targetList)
    if nextAction = SEGMENT then
        scene ← Divide(scene, targetList)
    else if nextAction = REASON then
        scene ← Comprehend(scene)
    end if
until nextAction = STOP
return state
```

scene has the structure described in 3.2.2.2 for a segment. The main process *Visual Perception* handles the special case, in which the segment list has only one segment, the whole scene. *Visual Perception* transforms an image $\mathcal{I}$ into a collection of information called *state*. *previousState* contains information collected from previous images. *machine* has the different modules, the actuator *Recognition* with the recognizers representing the known recognizable categories, the actuator *Segmentation* with the segmenters representing different ways of relating pixels and the rational module with all the known categories and their relations.

Process 2 Categorize a segment list

Require: $\mathcal{S}$, the segment list of an image or region
Require: $\mathcal{V}$, a list with all the categories that can be recognized
Require: *machine*, recognition, segmentation and rational modules

```
for each s ∈ 𝒮 do
    s ← Recognize(s)                          ▷ Modify η in s categoryList
    for each i ∈ 𝒱 do
        rate ← v(s, i)
        if rate isBetterThan max then
            max ← rate
            selectedSegment ← s
            selectedCategory ← i
        end if
    end for
end for
Activate(selectedSegment,selectedCategory),   ▷ Modify activationId
return selectedSegment
```

Ensure: *selectedSegment* and *selectedCategory* have been assigned a value

Categorize transforms a whole divided into parts, a segment list, into a categorized segment. *Categorize* modifies the η values in the selected segment *categoryList* and activates the selected category. Selection is a function of η, β and π as shown in equation Eq.1.

Process 3 Divide a segment by creating a list of subsegments

Require: *segment*, a segment with *activationId*, *categoryList* & *segmentList*
Require: *targetList*, a list with target categories
  segment ← Segment(*segment*) ▷ Modify *segmentList*
  segmentListCopy copy(*segmentList*)
  repeat
    s ← Categorize(*segmentListCopy*)
    repeat
      nextAction ← SelectAction(*machine*, *segment*, *targetList*)
      if *nextAction* = SEGMENT **then**
        s ← Divide(*s*, *targetList*) , *s* is a part of *segment*
      else if *nextAction* = REASON **then**
        segment ← Comprehend(*segment*) ▷ Modify β in *categoryList*
      else if *nextAction* = select another subsegment **then**
        pop(*s*, *segmentListCopy*)
      end if
    until *nextAction* = select another subsegment or return
  until (*nextAction* = return) or (*segmentListCopy* is empty)
  return *segment*
Ensure: *segmentList* is not empty

Divide is the recursive process in charge of building the hierarchy of segments that compose a whole. Each segment can be divided in subsegments until the selection function determines that further segmentation is not worth it. After the categorization of each subsegment, the system may choose to *comprehend* the subsegments categorized so far. The process *Divide* may stop when the system considers that it has gathered all the information that was required or if it considers that exploring other segments might be more useful. The algorithm contemplates the possibility of segmenting a region several times.

Process 4 Comprehend a segment

Require: *segment*, a segment with *activationId*, *categoryList* & *segmentList*
Require: $\mathcal{Z}$, a list with all the categories known
Require: *machine*, recognition, segmentation and rational modules

```
S copy elements of segmentList with activationId
T ← Combine(S)                    ▷ Create all the possible tuples with elements of S
for each t ∈ 𝒯 do
    segment ← Reason(segment,t)                    ▷ Modify β in categoryList
    for each i ∈ 𝒵 do
        rate ← u(t,i)
        if rate isBetterThan max then
            max ← rate
            selectedCategory ← i
        end if
    end for
end for
Activate(segment,selectecCategory)                 ▷ Modify activationId
return segment
```

Comprehend transforms a list of subsegments into a categorized segment. A selection function decides which combination of parts is more useful to define a whole. We have chosen to go over all the known categories, Z. This could require many resources. For many cases, considering the list of categories that can be recognized, V, may be a good option to reduce the computational cost. We could also build an ad-hoc category list for each problem following the idea of "knowledge pruning" (3.2.2.3).

3.2.2.5 Improving visual perception

Visual perception systems can be improved with better or more knowledge, new categories or relation between categories. Better definitions allow more reliable categorizations and are usually the result of better feature selection. New definitions allow categorizations that previously were not possible. When a perceptual

system tackles a new object or situation it must be able to learn it, to create a new definition so that when facing the same scene again it can categorize it.

Learning is an important branch of IA systems (Russell & Norvig [2014] Chapter V). Machines can learn from examples or learn from what they already know, by reasoning. Reasoning is the process by which the perceptual system passes from two or several propositions, called the premises, antecedent or prior knowledge, to another proposition, called the conclusion or consequent (Wallace [2011] p.20). Very often prior knowledge is represented by formal languages (3.2.1.3, Russell & Norvig [2014] Section 19), and can therefore be introduced by human experts.

Learning from examples can be unsupervised or supervised by a human or another system. In supervised learning the system observes input-output pairs, whereas in unsupervised learning no feedback is provided. But "How can we be sure that our learning algorithm has produced a hypothesis that will predict the correct value for previously unseen inputs?" (Russell & Norvig [2014] p. 724). We can easily adapt this question to the visual perception problem: "How can we be sure that our categories' definitions are good enough to characterize every object member of the category and only those when there are unseen representations of objects?". This is an example of PAC (Probably Approximately Correct) learning, which are based on the axiom "future examples are going to be drawn from the same fixed distribution as past examples". It is exactly the theory defended by Purves & Lotto [2003] (2.2.5) to explain why we see what we do.

Purves & Lotto [2003] suggests that rule-based schemes of vision are not able to deal with the inherent ambiguity of visual information. We would suggest that rule-based definitions are sometimes less appropriate than definitions learned from examples. Which would be the rules to differentiate between running an evacuating? Which would be the rules to differentiate the effect of one flash of lightning from another? One way of improving visual perception systems is to find definitions learned by examples, what we called knowledge by acquaintance to replace or complement definitions based on propositional knowledge. Categorization based on geometric evidence does not need to segment and comprehend, and is therefore more efficient.

The process of learning could easily be integrated in the algorithm described

in the previous section (3.2.2.4). The two functions *Initialize* and *Activate* represent learning. With the first one *status* acquires knowledge about the problem coming from previous images. The second one could be associated with a new action *LearnFromExample* by which the *machine* acquires new recognition capacities. Once that a segment has been categorized (*activated*) the system acquires knowledge by acquaintance from the image. For this purpose categorization could be seen as the process providing labels for the images so that supervised learning can take place. Let's remember that the same segment might be activated more than once and therefore might contribute to the learning process of different recognizers.

3.3 Analysis of implementation methods of visual perception systems

There are many different approaches to implement segmentation, recognition, reasoning, selection or learning. In this section we present some of the most important ones and review relevant methods that use them. We have organized the section in five subsections, one per type of computation.

3.3.1 Segmentation

Segmentation is the process of dividing an image in parts, regions or segments. These segments can play two different roles in visual perception. They can represent the target categories, or provide information to categorize the whole. A segment is a set of pixels that are related, and thus can be seen as a category. The elements of this set are members of the category while the rest is not. Segmentation can threfore be seen as a process of categorization of an image, by which each pixel is assigned to a particular set. Considering a segment as a category may not be intuitive because by itself it is only intrinsic information that may or may not be known by the system. Usually we feel more comfortable when categorization yields known categories, categories with signification. What makes the difference between categories with signification and without it, is the fact of relating the set of properties defining them to extrinsic information.

However the difference between the kind of categorization used to recognize an object and the one used to divide an image is the approach to the constraints defining the categories. *Recognition* evaluates how well the elements satisfy each of the definitions known by the system, while *segmentation* evaluates which elements satisfy a given definition. We have divided the definitions used for segmentation in three classes depending on the main constraints. In the first class, elements belong to a category when they satisfy a geometric constraint about their coordinates. In the second class, semantic segmentation, elements belong to a category when they satisfy geometric constraints of surfaces. In the third class, elements belong to a category when they satisfy a spatio-temporal condition. The following subsections present some of the most relevant methods in object recognition literature and analyze their segmentation approach.

3.3.1.1 Sliding window

A straight forward way to divide an image is to consider it as a grid of smaller images. The size of each segment of the grid determines the number of segments, smaller segments means more segments. If we need to find objects of different sizes we have to apply different segment scales. In order to avoid missing objects all the positions must be processed. The approach that, for all positions and scales in an image evaluates a score function to find its local maxima, is referred to as "sliding window" (Harzallah *et al.* [2009]). Sliding window has been successfully implemented to detect human bodies (Dalal & Triggs [2005]), human faces (Viola & Jones [2004]) and different objects (Laptev [2006]) from the PASCAL database (Everingham *et al.* [2010]).

The sliding window algorithm itself is very simple but since the search space is huge, the number of window candidates (segments) can be very large. As a consequence potential algorithms for the recognition module have to be selected under strong limitations from a performance point of view. Viola & Jones [2004], for example, uses a cascade of weak classifiers to improve performance. To limit the amount of window candidates, exhaustive search can be limited by using heuristics to guide the search. Lampert *et al.* [2008, 2009] suggest to use a *branch and bound* approach. Alexe *et al.* [2012] describes cues to measure the objectness

of an image windows. Most successful methods based on sliding window follow a top-down approach to minimize the amount of segments generated and evaluated. Harzallah *et al.* [2009] states that the performance of sliding window based systems depends, among others, on an efficient search strategy.

In this approach the criteria used to generate segments are position and scale. The elements of the segment do not have to satisfy any other constraint, so the segmentation process gives no information about the data bounded by the window and no information about the relations between the parts of the image. In the previous section we have discussed the importance of the relations between the parts and how these relations constitute valuable information for perception. Felzenszwalb *et al.* [2010] presents a method built on sliding window (Dalal & Triggs [2005]) and completed with a mixture of multiscale deformable part models. The use of part based models improves the precision of object detection. The authors suggest that future work could include grammar based models that represent objects with variable hierarchical structures.

3.3.1.2 Semantic segmentation

Gibson [1986] claims that surfaces are one of the basis of perception (2.2.3). Hoiem *et al.* [2007] recovers Gibson's ideas about surfaces and proposes a method to construct the surface layout, "a labeling of the image into geometric classes". Felzenszwalb & Huttenlocher [2004] proposes to consider the pixels of an image as vertices of a graph, where the weight of the edges is some measure of the dissimilarity between the two pixels connected by that edge. The weights define relations between the pixels, they define the intrinsic structural information of the image. However not all the relations are relevant, segmentation algorithms should select the ones that are likely to be given signification. Felzenszwalb & Huttenlocher [2004] chooses measures based on the difference in intensity (color) to segment images into regions, similar to what we could call surfaces.

Beyond defining surfaces, segmentation should define figure and background. This topic was widely covered by Gestalt theorists (2.2.4). In the work Carreira & Sminchisescu [2010], Gestalt properties are used besides graph partitions and regions properties to predict whether segments have regularities typical of pro-

jections of real objects (figures). Endres & Hoiem [2010] uses three classifiers to predict if a region is likely to be foreground or background, if two regions are likely to lie on the same object and if a region lies on the left, right, top or bottom of an object. Then a ranking model ranks the likelihood of a set of proposals of being an object. In Carreira & Sminchisescu [2010] and Endres & Hoiem [2010] the methods require previous training but are category-independent, this means that segmentation does not depend on the object represented by the segment. Category-independent approaches follow the idea of early selection (2.3.3), selection comes before recognition. Attention would be directed to the candidates ranking higher.

Since semantic segmentation processes the intrinsic structure of an image, it may seem natural to have category-independent algorithms. However category-independence is not exclusive of semantic segmentation, we already presented the work Alexe *et al.* [2012] based on sliding windows, which is category-independent. This is achieved by creating a generic category in which all the different objects may fit, with a constraint called "objectness". "Objectness" is somehow what Rubin [1958] defines with "richer, with a more differentiated structure, with greater structural solidity of the color and appear to be closer to the viewer than the field experienced as background" (2.2.4).

Semantic segmentation can also be category-dependent. Arbelaez *et al.* [2012] uses multiscale low-level hierarchical segmentation (Arbelaez *et al.* [2011]) to produce "high quality object candidates... in a simple and generic way without mid-level information or learning" and then applies a multi-class high level region representation that integrates scanning-window part detectors and global appearance cues (Felzenszwalb *et al.* [2010]). This representation is used to make pixel level decisions, in other words, to label each pixel. We have to note, that each pixel may belong to more than one region, since region candidates have been produced by a multiscale hierarchical segmentation process.

Uijlings *et al.* [2013] implements Felzenszwalb & Huttenlocher [2004] using multiple thresholds and a hierarchy, showing the latter better results. The authors state that "images are intrinsically hierarchical [...] This prohibits the unique partitioning of objects for all but the most specific purposes". The paper shows how an image region is formed because of a variety of reasons, similar color,

texture or inclusion, and therefore suggests the use of diverse strategies to find objects. The use of a hierarchical algorithm is also a way to take into account all object scales, aiming to capture all possible object locations. Uijlings *et al.* [2013] is an example of several ideas exposed in this dissertation. The elements of an image can be related in a variety of ways, and among the resulting relations, more than one can be given signification. The more "strategies" the perceptual system can execute, the more chances to find objects. Depending on the chosen configuration, "single strategy ", "fast selective search" and "quality selective search", the number of strategies used, windows created, time consumed and accuracy increases. The results of segmentation depend not only on the algorithm, but also on how it is executed. In fact more than one configuration could be applied, starting by the fastest. Its results may or may not advice to process the image with a more powerful configuration. "Quality selective search" takes 20 times more time than "fast selective search", but the recall increases from 0.98 to 0.99. The idea of starting with a fast processing configuration is justified. Another interesting point is the fact that the process of grouping regions is repeated until the whole image becomes a single region. To comprehend the essence of the image, it is not enough to grasp the essence of its parts, a relation between all the parts must be apprehended to grasp the essence of the whole. Girshick *et al.* [2014] uses Uijlings *et al.* [2013] to generate category-independent region proposals, which are processed with a convolutional neural network (Krizhevsky *et al.* [2012]) to provide more precise object location than sliding-window approaches.

3.3.1.3 Motion detection

The previous segmentation types process still images. However motion plays an important role in perception. In section 2.1.3 we present works suggesting that the human visual system has different pathways to process information about spatial structure (P pathway) and information about temporal changes (M pathway). Motion detection can be used to divide images into foreground (moving elements) and background (still elements).

Background subtraction techniques are probably the most popular choice in the literature to detect motion. The idea is to extract foreground objects from

an image by subtracting a "background model" image from the original one. The main challenge is to generate a "background model" fast and with robust results. Brutzer *et al.* [2011] and Bouwmans [2014] describe the main challenges for background subtraction (BS) methods. Piccardi [2004] compares different background subtraction methods. "Running gaussian average" has the best speed performance while "Mixture of Gaussians" or "Kernel density estimation" give better accuracy. Xu *et al.* [2016] classifies background modeling methods in parametric and nonparametric categories . Two methods show a performance over 20 FPS. "Adaptative Gaussian Mixture Model" (AGMM) improves classic Mixture of Gaussians by automatically adapting to the scene by choosing the number of components for each pixel (Zivkovic [2004]). "Visual Background Extraction" (ViBe) improves other methods by storing values of pixels taken in the past and choosing randomly which values to substitute instead of replacing the oldest (Barnich & Droogenbroeck [2011]). Both outperform other methods in difficult conditions, such as bad weather.

The second approach, temporal filtering, is based on temporal differencing (Lipton *et al.* [1998]). This method uses a thresholded difference of pixel between consecutive images (two or three) to extract the moving object, so it shows high computing performance. However its detection accuracy may be weak, failing in extracting all the relevants pixels of a target object or leaving holes inside moving objects (Kim & Street [2004]).

Finally optical flow is an approximation to image motion defined as the projection of velocities of 3D surfaces points onto the imaging plane of a visual sensor (Beauchemin & Barron [1995]). Different optical flow techniques are detailed in Barron *et al.* [1994], most of them are computationally complex. Another important withdraw is that optical flow algorithms are very sensitive to noise (Hu *et al.* [2004]).

3.3.2 Recognition

Recognition is the process by which the perceptual system evaluates whether the elements of an image satisfy the constraints of a set of known categories. The challenge is to find the best definition for each category. In section 3.2.1.2 we

show several branches of geometry providing tools to represent the geometric properties of an object or image. In this section we present the features used by some of the most relevant methods used in computer vision. We have made two classifications, one to differentiate low and high level features and a second one to differentiate local and global features. We finally review algorithms used to classify the features.

3.3.2.1 Low level and high level features

One of the main concepts in geometry is curvature. Nixon & Aguado [2012] considers curvature "as the rate of change in edge direction", which characterizes the points in a curve. Points where the edge direction changes rapidly are corners, whereas points where there is little change in edge direction correspond to straight lines. According to Nixon & Aguado [2012] these extreme points are very useful for shape description and matching, since "they represent significant information with reduced data" (p.180). Indeed, under these definitions, the difference between a pixel representing a corner and one representing a straight line is the relation between a pixel and its neighbors.

A surface can be represented by a set of its invariant properties (3.2.1.2). Invariants are one of the tenets of Gibson [1986]. According to this work, in order to perceive persistence and change we pick up invariants of the structure of the ambient optic array. In computer vision, using invariant features to represent things is important to be able to recognize this thing in different environments. When a feature depends on the object's position or illumination conditions, we have more chances to miss that feature.

All the previous features are low level features, features that can be extracted without any shape information (Nixon & Aguado [2012] p.138). Shapes are particular spatial relations of pixels, and are considered as high level features (p.218), for example, a face. We could think of high level features as the ones, that by themselves have signification, whereas low level features are the ones that have not. Maybe the most important idea behind low and high level features, is once again, hierarchy. High level features are relations of low level features.

The values of the pixels of an image are likely the simplest features of an image,

the relation between the coordinates of a pixel (x, y) and its value. Relations between these features are high level features called templates. Spatial relations can be directly approached by model or template matching. A template can be represented by a function $T(x, y)$ mapping coordinates (x, y) of a window (image segment) to some value. We have seen that an image can be represented by a function $I(x, y)$, we could therefore consider template matching as a method of parameter estimation. More details can be found in Nixon & Aguado [2012] (p.222-230). Since a template is a function over a window, template matching has difficulty to deal with rotation and scale invariance. Solutions in the frecuency domain have been tried to deal with these difficuties (Derrode & Ghorbel [2001]), but still face one of the main issues for template matching: processing speed. Stockman & Agrawala [1977] shows that Hough curve detection can be equivalent to template matching and Princen *et al.* [1992] suggests ways of implementing Hough-like algorithms to improve performance. Weiss *et al.* [2012] demonstrates that Hough transform is well suited for real-time detection.

The relation between the coordinates of a pixel and its value are just one among many. For example, Viola & Jones [2001] implements a cascade of simple low level features, Haar wavelets, to represent shapes. Cascade filtering is useful to minimize the cost of extraction of the features. More expensive operations are applied only at locations that have already passed filters with lower processing costs. A filter is just a binary function that classifies a set of pixels, ergo a characteristic function (constraint) of a category (3.2.1.2). We find the idea of cascade filtering in the very popular work Lowe [2004] presenting Scale-Invariant feature transform. In this case the author uses distinctive scale-invariant keypoints instead of simple templates like Haar wavelets. "Descriptors" of the keypoints are compared to recognize the searched thing. "Descriptors" define the constraints that a region must satisfy to be identified as a particular category. Extracting more than one keypoint increases the probability to match an object correctly.

Schneiderman & Kanade [2004] presents a method based on two pre-computed probability distributions representing the statistical knowledge of the object appearance. Probabilities are computed over different parts of the object and then combined in a classifier. In the paper we find some of the main points of our theory, "parts need not have a natural meaning to us (such as a nose or an eye),

but could be defined as a group of pixels, or transform variables, that satisfy certain mathematical properties", such properties are low level features.

Today some of the most popular methods for image recognition are based on Convolutional Neural Networks (CNNs). CNNs already had good results in the 1980s, for example recognizing handwritten numbers (LeCun *et al.* [1989]), but today's popularity comes from the results showed by CNN trained with 1.2 million images from Imagenet database (Deng *et al.* [2009] Krizhevsky *et al.* [2012]). CNNs have an architecture with different types of layers, where convolutional, pooling and fully-connected layers are usually implemented. Convolutional layers use kernels, which usually are small templates made with a set of weighting coefficients. Template convolution calculates new pixel values by placing the template at the point of interest, multiplying surrounding pixels by the weights and summing the results. Template convolution is therefore a relation between a set of pixels. Like other approaches combining low level features, CNNs use a classifier to represent the constraint satisfaction evaluation that categorizes an image. CNNs are likely the most important instance of "Deep Learning" algorithms (LeCun *et al.* [2015]), which have structures built with several layers of processing units. The different layers relate information from the previous ones, and therefore can be seen as a hierarchy of features, from lower to higher.

3.3.2.2 Local and global features

In the previous section we have seen how features can be organized in a hierarchy where only high level features are given signification. The idea of hierarchy is also found in Hall [1979] with a proposal about the organization of scenes, and thus images: "natural scenes may be described in terms of hierarchical structures such as scene-object-surface-boundary-point in which each pattern is described in terms of simpler patterns". This hierarchy is represented with local and global features. Local features are features from parts of a whole, while global or holistic features are features of the whole. "Cars" may be features of a "parking", "wheels" may be features of a "car", "circular" may be a feature of a "wheel", such that a parking could be recognized by its local features, "cars" , the object "car" could be recognized by its local features "wheels" and the object "wheel"

could be recognized by the global feature circular.

Bottom-up approaches integrate the categorizations of the parts to categorize a whole. On the other hand global features categorize the whole without explicitly categorizing the parts. Oliva & Torralba [2001] claims that object recognition is not needed to recognize a scene, that the "gist" (essence) of a scene can be grasped by the means of global features (Oliva & Torralba [2006]). Global features are the result of relations between all the elements composing a whole. An image, a matrix of pixels, is a global feature of the image itself. Such a feature would be too specific, so that in order to define categories more generic features should be chosen.

Moments are quantitative measures of the shape of a set of points. Moments were originally introduced in image analysis by Hu [1962] and further developed by Teague [1980] with Zernike moments (Nixon & Aguado [2012] p.383-393). Torralba & Oliva [2003] presents how simple image statistics can be used to predict the presence or absence of objects in the scene before exploring the image, thus without segmentation nor object recognition.

Techniques used to describe regions can also be applied to describe the whole image. One of the important characteristics used to identify regions is texture (Haralick *et al.* [1973]), an image could be considered as a texture, such that texture descriptors (Haralick [1979]) represent the image. Ojala *et al.* [1996] compares different texture measures and suggests that distributions of features values should be used instead of single values. Ojala *et al.* [1996] also present texture measures based on local binary patterns (LBP), which combine several local descriptions of the whole image into a global description. LBP have been successfuly used in a wide range of applications, like face recognition (Ahonen *et al.* [2006]) or writer identification (Bertolini *et al.* [2013]).

Lazebnik *et al.* [2006] suggests that global features can not only be used to capture the "gist" of an image but also to inform the subsequent search for specific objects. An image is repeatedly subdivided to compute histograms of image features over the resulting subregions. He *et al.* [2004] suggests to extract features at different scales and combine them using "Conditional Random Fields" (CRF - Lafferty *et al.* [2001]) to help disambiguate classifications. The idea is that context can provide useful information to correctly categorize a part. In

the same line Murphy *et al.* [2003] and Murphy *et al.* [2006] present how global features can be used to help resolving local ambiguities. It is somehow similar to what we find in Gibson [1986] "ground theory of space perception" where the character of the visual word is not given by objects but by the background of the objects (p.150). Oliva & Torralba [2007] concludes that:

> a scene composed of contextually related objects is more than just the sum of the constituent objects. In the absence of enough local evidence about an object's identity, the scene structure and prior knowledge of world regularities might provide the additional information needed for recognizing and localizing an object. Even if objects can be identified by intrinsic information, context can simplify the object discrimination by decreasing the number of object categories, scales and positions that must be considered. How objects are remembered also depends on the scene context they are in.

3.3.2.3 Classification of features

We say that a model m, in this case an image, satisfies a sentence α, in this case a property, if the sentence α is true for this model. How could a measure be true or false ? Again we can use a function mapping the measure to the value true or false, or to the probability of truth or falsity. Such a function is called a classifier. A simple way to implement a classifier is to introduce a threshold, measures above it are attributed one class and the ones below it another one. In the previous subsection we have seen several examples of methods using classification. In all of them more than one feature was used. Perception is about relations, in this case relations of features. We find different approaches in the literature to implement classifiers, being neural networks and support vector machines among the most popular choices (Forsyth & Ponce [2003] p.601-618). Both are examples of parametric models.

Parametric classifiers have a finite number of parameters, a better choice of the parameter set yields better classification results. In section 3.3.5 we discuss how these parameters can be chosen. There is another type of classifiers that does not use parameters, but the idea of distance (Nixon & Aguado [2012] p.417-420).

"It is reasonable to assume that example points 'near' an unclassified point should indicate the class of that point" (Forsyth & Ponce [2003] p.587). Nearest neighbors methods are based on this heuristic. In parametric models, labeled samples are required to determine the parameter set, in the non-parametric models, they are also required to calculate the distance between them and the evaluated image. Knowledge for classification can either be represented with the parameters of parametric classifiers or with the samples used in non-parametric classifiers.

When the spatial relation between features is relevant, in other words, when context matters, graphical models like Markov Random Fields or Conditional Random Fields (Lafferty *et al.* [2001]) are useful because they have the capacity to predict sequences of labels. Conditional Random Fields have successfully been used in some promising methods to improve the results of CNNs (Farabet *et al.* [2013] Chen *et al.* [2015]).

Classification evaluation. Let A be the predicate *IsTheObject(x)* and B *Property(x)*, when $p(A \wedge B)$ is different from $p(B)$ some elements satisfying B do not satisfy A. This means that some object x with Property(x), does not belong to the category defined by IsTheObject(x). Such cases are called False Positives. On the other hand it may happen that an object x belonging to the category does not satisfy Property(x), we call such cases False Negatives. In order to evaluate classifiers different metrics can be used, some of the most important are recall, precision, F-measure and percentage of correct classification:

Recall	$R = \frac{TP}{TP + FN}$
Precision	$P = \frac{TP}{TP + FP}$
F-measure	$F1 = \frac{RP(1+\alpha)}{R + \alpha P}$
PCC	$PCC = \frac{TP + TN}{TP + TN + FP + FN}$

Table 3.1: Evaluation metrics

Metrics depend either on False Negatives (FN), False Positives (FP) and True Positives (TP) or True Negatives (TN). As one may guess a True Positive is an

object x belonging to the category with the label of the category. A True Negative is an object that does not belong to the category with a label different from the one of the category. More information about evaluation of classifiers can be found in Powers [2011].

3.3.3 Reasoning

When an image or one of its regions is categorized the perceptual system can reason about it. Reasoning is the process by which information unknown by the agent is inferred from information it knows. The information known by the agent is the information that has been gathered so far and the information in its knowledge base (KB). The information that can be inferred are probabilistic propositions, that say something about the image or its regions. Reasoning is one of the main topics in any AI book (Russell & Norvig [2014] Chapters III & IV, Nilsson [1998] Chapter III, Pearl [2014]). In this section we introduce expert systems and present relevant methods to handle categories and belief.

3.3.3.1 Expert systems

For many years expert systems have been used to support activities based on specific knowledge: agriculture, communications, construction, financial, manufacturing, transportation or medical (Feigenbaum *et al.* [1989]). The knowledge base of an expert system represents knowledge using some kind of formal language like First-Order Logic (FOL) (Barwise [1977]), so that an inference engine can manipulate that knowledge and deduce information requested by the user.

Such a system requires of knowledge engineers, computer scientists with artificial intelligence training, to represent the knowledge from a human expert in a form that can be entered into the knowledge base (Nilsson [1998]). This conventional approach to knowledge acquisition faces several limitations, theoretical and practical (Potter [2003]). Experts should be able to provide the actual knowledge used in a task but this may not always be the case, either because some kind of knowledge is hard to express or because the expert is not able to retrieve it. On the other side, knowledge engineers may also misunderstand, misinterpret or fail to grasp the domain in hand. "To some extent, knowledge engineering is an art,

and some people become more skilled at it than others" (Winston [1993]).

This weakness in expert systems is specially relevant for visual perception. It is the same problem that we presented in section 3.2.1.4 with the quote from Austin & Warnock [1964]. How can a cat in any position be described such that no other animal can be mistaken for it? It seems difficult to find a set of propositions expressing a definition to answer this question. On the other hand Simonyan & Zisserman [2015], without any propositional knowledge, achieves impressive results to classify cat among other animals under different light illumination and poses.

3.3.3.2 Natural language

However propositional knowledge is important. Most of the humankind's knowledge has been conceptualized in dictionaries, encyclopedias, books, journals, newspapers etc., an increasing number of them are digital or have been digitized (Coyle [2006]), and hence are available in computer networks. Making computers understand natural language is the key to unlock human knowledge about the world.

Several approaches have been used to describe natural language (NL), a classic one are production rule systems (Chomsky [1956]). A production system consists of a set of rules, a working memory and a long-term memory. Its basic operation runs repeatedly through a cycle of three processes: recognize, resolve and act. "Recognition" matches rules against the current state of the working memory, which results in the "conflict set". "Resolve" selects a suitable set of rules from the "conflict set" to execute. "Act" executes the actions and updates the working memory of on-going assertions (Brachman *et al.* [1992]).

A second approach is the ontology web language (OWL). OWL facilitates greater machine interpretability of Web content by providing additional vocabulary along with formal semantics. OWL can be used to explicitly represent the meaning of terms in vocabularies and the relationships between those terms (McGuinness *et al.* [2004]). OWL is an XML-based vocabulary for describing properties and classes, among others, relations between classes, cardinality, equality, characteristic of properties or enumerated classes. The strict definition of

static structures given by XML schemas limits the capacity of OWL to represent beliefs, knowledge that contains subjective degrees of confidence.

A better representation for beliefs are Bayesian networks, also called "belief networks" (Pearl [2014]). A Bayesian network is a probabilistic directed acyclic graph (DAG) whose nodes represent random variables and whose edges represent conditional dependencies, providing means to express joint probability distributions over many related hypotheses. However Bayesian networks have a limited expressiveness equivalent to propositional logic, and therefore are not suited to refer to objects in the world (concepts), unlike First-Order Logic.

Semantic networks have an expressive power equal to First-Order Logic. Sowa [2006] presents six common kinds of semantic networks : definitional, assertional, implicational, executable, learning and hybrid. Definitional networks emphasize the *IsA* relation between a concept type and a newly defined subtype, such that any subtype inherits the properties of the supertype. While the information in definitional networks is often assumed to be true, information in an assertional network is asssumed to be contingently true, which makes them suitable to represent the conceptual structures underlying natural language semantics. An implementation of semantic networks to represent general knowledge and how it is expressed in natural language is ConceptNet (Havasi *et al.* [2007]; Liu & Singh [2004]; Speer & Havasi [2012]).

3.3.3.3 Language and perception

A perceptual system performing image description can be seen as a "visual translator" (Herzog & Wazinski [1994]) which generates natural language expressions from images. Besides being the output of a perceptual system, language expressions can also be an input. Srihari [1994] classifies computational models for integrating linguistic and visual information in two areas based on the input types used by the systems as well as their functionality. The first group are the systems that accept either language or visual input, but not both, while the second group are the systems that deal with both linguistic and visual inputs. Bernardi *et al.* [2016] sorts computational models for image description in three categories. The first one called "direct generation" uses information detected in the image like ob-

jects, scene types, actions etc. to drive natural language generation. The second category referred to as "retrieval in visual space" exploits similarity between images in the visual space to transfer descriptions of known images to query images. The third category is also based on retrieval but in this case similarity is computed over the visual and linguistic space, thus called "retrieval in multimodal space". Bernardi *et al.* [2016] reviews 35 different approaches to automatic image description published in less than five years, 17 of them retrieve descriptions and images from a multimodal space.

3.3.4 Selection

This dissertation analyzes visual perception as a search problem with a state space that in many cases cannot be approached by an exhaustive strategy. Selection is therefore a key element that enables the possibility of finding target states with limited resources. In this section we review how relevant methods implement selection techniques.

3.3.4.1 Cascade methods

Cascade based methods process each segment of an image with a weak classifier to get evidences about the relevance of what is represented on it. A weak classifier should eliminate a large number of regions, those with low evidence to represent a target category, with very little processing. The objective is to minimize the number of False Negatives, interesting regions classified as non interesting, even if the number of False Positives, non interesting regions classified as interesting is high. Then a more complex classifier is used to eliminate more candidate regions. This approach is followed by well known methods (Viola & Jones [2001, 2004]). For these methods the decision that has to be taken is whether the region should be discarded or if more features should be extracted and classified. Each segment is classified at least once. When cascade-based methods are implemented with sliding window segmentation, the number of segments can be controlled by scale constraints. If the system has information about the size of the object it can eliminate a range of scales, such that less regions are generated.

3.3.4.2 Branch and Bound

Forecasting the size of the target is not realistic in real world problems. Lampert *et al.* [2008] proposes to avoid wasting resources evaluating all candidate regions and target the search directly to identify the regions with highest scores from a quality function, regions which are likely to represent a target category. Efficient Subwindow Search (ESS), "organizes the search over candidate sets in a best-first manner, always examining the rectangle set that looks most promising in terms of its quality bound". Only the most promising rectangle set is split into a subset of rectangles which are evaluated. Branch and bound methods discard, and therefore select regions depending on a quality function. Alexe *et al.* [2010] shows that the number of segments evaluated by ESS (Lampert *et al.* [2008]) can be very large when non-linear classifiers are used to make the selection.

3.3.4.3 Selective search

On the other hand *selective search* described in Uijlings *et al.* [2013] proposes to select a combination of diverse similarity measures. Depending on this combination more or less segments are generated. The work presents three examples called "single strategy" with 362 regions and 1 strategy, "selective search fast" with 2147 and 8 strategies and "selective search quality" with 10.108 regions and 80 strategies. An agent implementing *selective search* could decide which combination seems more appropriate at each moment, and for each segment. Another strong point about this method is that segmentation is bottom-up or data-driven. Starting from an initial group of regions created with the method described by Felzenszwalb & Huttenlocher [2004] a greedy algorithm iteratively groups regions together. Instead of selecting a range of scales, this method suggests to select a combination of similarity measures to divide an image or region into smaller segments.

The selection can be knowledge-driven. Instead of starting with a "quality" search the agent might start with a faster combination that generates less segments, then a categorization of these segments could provide information suggesting further segmentation over one or the other region. A method implementing *selective search* combined with a top-down approach is described in Xiao *et al.*

[2015].

3.3.5 Learning

Learning is the process by which visual perception is improved. In section 3.2.2.5 we state that visual perception can be improved with better definitions, with new categories or with information about them. Better definitions can be achieved with better features or better constraints. New definitions can be created by associating a set of properties and conditions with other categories. Information about categories are relations between them. In this section we present methods to improve or learn new definitions for categories and then analyze how perception can increase the knowledge of the system and how knowledge can be used to improve perception. Finally we show how state-of-the art learning techniques can easily be fooled.

3.3.5.1 Improving recognition

Learning parameters. In section 3.3.2.3 we present two types of classifiers, parametric and non-parametric. Parametric classifiers depend on a set of parameters to attribute a class to a feature vector. In order to find the best parameters, methods like neural nets and support vector machines require training. The process of training could be seen as one way of implementing knowledge by acquaintance. Previously labeled samples (known samples - *kennen*) are used to adjust the weights (parameters) of a neural net (LeCun *et al.* [1998]) or a support vector machine (Vapnik [2013]). These techniques are examples of supervised learning. Each time that a perceptual system categorizes a region, the result can be used to train, and therefore improve the recognition module.

Learning features. Good parameters can be learned, but also good features. Chandrashekar & Sahin [2014] presents several methods for feature selection and show how "more information is not always good in machine learning applications". Instead of relying on a set of features selected by a human, machines can do it by themselves. Farabet *et al.* [2013] proposes a multiscale convolutional network trained from raw pixels for scene classification. Texture, shape and con-

textual information are successfuly captured without need of engineered features. Grangier *et al.* [2009] and Pinheiro & Collobert [2014] also avoid "hand-crafted features" and fed directly the neural network with the pixels of the image. The multiscale approach is substituted by sequential series of convolutional networks (LeCun *et al.* [1995]). Pinheiro & Collobert [2014] shows that a recurrent architecture (Recurrent Neural Networks - RNN) can also capture texture, shape and contextual information. The authors claim that their method is "simpler and completely feed-forward, as it does not require any image segmentation technique, nor the handling of a multi-scale pyramid of input images". While Farabet *et al.* [2013] and Chen *et al.* [2015] include CRF to increase the capability of modeling global relationships or improve localization, Pinheiro & Collobert [2014] avoids any graphical model in order to keep simplicity and reduce computing costs. Finally Zheng *et al.* [2015] proposes to formulate CRF as an RNN to form part of a deep network to perform end-to-end training combined with a CNN and achieve state-of-the-art on Pascal VOC segmentation benchmark (Everingham *et al.* [2010]).

Supervised learning depends on the availability of labeled data. For some perceptual systems this might not be easy. Unsupervised feature learning is a methodology in machine learning to build features from unlabeled data. A common approach is to use an encoder-decoder architecture. A function called encoder generates a feature vector from an input, in this case an image, then another function called decoder reconstructs the input from the feature vector. The reconstruction error is the loss function to train the encoder and decoder as parametric classifiers, searching for the best parameters to minimize the reconstruction error. Clustering algorithms like Kohonen (Kohonen [1990]) or K-Means (Jain [2010]) can be seen as unsupervised learning algorithms, where the index of the node or cluster is the feature. Dimensionality reduction algorithms, like PCA (Forsyth & Ponce [2003] p.596) can also be considered as unsupervised learning algorithm. In this case the weights associated with the eigenvectors would be the features.

Clustering or dimensionality reduction methods can be used directly over images and achieve low error in recognizing the same image, but very often the features are not robust to variations. For most of the real world applications,

invariant features are required. In section 3.3.2 we present several methods to extract invariant features (Viola & Jones [2001], Derrode & Ghorbel [2001]). However those methods use "hand-crafted features". Ranzato *et al.* [2007] proposes an unsupervised method for learning sparse hierachical features (low level) that are invariant to local shifts and distorsions, directly from the raw pixels. This method is presented as an alternative for supervised learning method for situations where the lack of labeled data causes over-fitting. More recently Le *et al.* [2013] achieves to build a face detector (high level feature) only from unlabeled images. Unsupervised learning can be seen as a way of increasing the capacity of the recognition module.

3.3.5.2 Knowledge and perception

We have said that with each categorization the recognition module can be trained and thus improved. When the parts of a whole are categorized the agent can also learn this relationship. It seems reasonable that when more examples of the same relation between the same set of parts and the same whole are found the agent has more evidences that the whole is made of these parts. When the amount of evidences if high enough, it could be considered a known proposition and therefore be included in the knowledge base. This could be seen as the transition from VSTM to long term memory. In classical IA, perception is a form of knowledge acquisition.

On the other hand knowledge about the categories can be used to improve perception, more precisely the processes of segmentation and recognition.

3.3.5.3 Fooling classifiers

Deep convolutional neural networks achieve impressive results recognizing objects in large datasets like Imagenet (Krizhevsky *et al.* [2012]; Simonyan & Zisserman [2015]). However these techniques, which represent state-of-the-art in object recognition can easily be fooled. Nguyen *et al.* [2015] shows how deep neural networks classify unrecognizable images (3.3) with a confidence above 99% .

We think that the experiments shown in Nguyen *et al.* [2015] support the idea

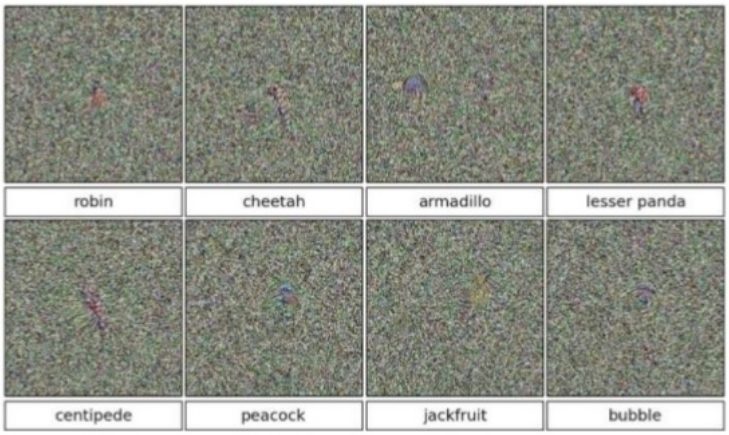


Figure 3.3: Unrecognizable images Nguyen *et al.* [2015]

that visual perception has to be approached as a process of information gathering, in which pattern recognition is one of the operations.

3.4 Summary

Visual perception is a process by which information is collected over time, so that categories can be computed. A category is a set of objects that satisfy a definition, which is a set of properties. Properties can be classified in two types intrinsic and extrinsic. Intrinsic information are the relations between elements of an image and can be represented by geometric features. Extrinsic information are relations between concepts called propositions. Propositions are the fundamental elements in formal languages, which are widely used to represent knowledge. However propositions are only good to represent the so called "propositional knowledge" in opposition with "knowledge by acquaintance". The latter is better suited to represent intrinsic information which is hard to express with words. Knowledge by acquaintance can be as reliable as propositional knowledge or even more.

An object is categorized as a category when it satisfies its definition. *Recognition* is the computation by which the perceptual system evaluates whether the subject satisfies the definition of a category. Recognition is a fundamental computation for visual perception but is not sufficient to emulate human vision. Images

might contain one or many objects, therefore a perceptual system must be able to divide an image into the parts that represent these objects. This computation is called *division* and might also be useful to generate features. The parts of a whole can be categorized in order to categorize the whole, in a computation that we have called *comprehension*.

The goal of visual perception is to find targets, to find categories. However visual perception should not be considered as a process of simple categorization, of pattern matching but as a process of information gathering, a search problem. The amount of possible relations between the elements of an image and between the categories of a knowledge base can create an unapproachable space of possible categorizations for a single image. Human vision results can only be achieved by a process of guided search.

We have chosen the intelligent agent paradigm to represent visual perception as a search problem. A rational agent whose goal is to find a set of categories would select at each moment the computation that maximizes the probabilities to reach its goal. Such computations are performed with the data from the image and the knowledge of the agent. The agent knowledge is composed of the prior knowledge and the percepts, information gathered from the image. We have proposed a program to implement the process of information gathering. The information collected defines the status of the system, which can be evaluated to find categories. The status of the system represents the Visual Short Term Memory (VSTM).

The proposed algorithm combines top-down and bottom-up approaches, in a similar way as Bundesen & Habekost [2008] TVA. The main process is a top-down computation generating a hierarchy of segments. The agent explores the hierarchy of segments following a strategy inspired by *branch and bound*. Branching is a recursive process of segmentation, from image to regions, then subregions etc. Bounding limits the number of regions that are segmented using heuristics about the probability of finding relevant information in the subregions. Each region can be categorized by classifying global features or by integrating local features. Categorizing with low level features avoids segmenting a region to extract the high level features associated to its parts.

Visual perception can be improved when the agent is able to learn. An agent

can learn from its knowledge or from examples. In the first case reasoning generates new propositions from premises. In the second case the agent creates definitions based on intrinsic information from the examples. Expressing the properties that define an object or category with propositions is often complicated. This fact combined with the stochastic nature of visual perception make definitions learned from examples and based on geometric information more efficient than ruled-based learning schemes for many real world cases. Moreover recognition of geometric information does not need to segment a whole into parts, and thus is more effective.

Chapter 4

Applications of machine visual perception

In the previous chapter we claim that the result of perception are categories, and categories can be defined with properties or features. In section 3.3 we present several methods to implement feature extraction. In this section we show that different types of categories, which can be the result of human vision, can also be perceived by machines. We also analyze the role of the three operators described in section 3.2.2: recognize, segment and reason in the process of perceiving the different categories, and the relation between the subprocesses *Categorize* and *Divide*.

Four experiments have been chosen for this analysis. The first one deals with crowd activity perception. It might seem challenging since crowds are made of many people and their behavior is sometimes hard to describe precisely. The second experiment is about handwriting authorship perception. Police corps have developed scientific methods to identify the author of a handwritten document by categorizing the characters inside. However handwritten documents are hard to segment into characters because writers generate touching pieces of ink and not separated characters. The third experiment deals with intrusion detection systems based on video surveillance. Like human attention, computer based systems might be attracted by motion, however the motion generated by sudden illumination changes is not useful information and challenges the efficiency of

the system. Finally, in the fourth example we analyze whether a computer can emulate human subjective opinion about an artistic expression: calligraphy.

4.1 Activity perception

Our first example is about activity perception in video surveillance installations and is based on the work Cermeño *et al.* [2013]. The number of video surveillance cameras is increasing year after year, making automatic event recognition the only way to manage the huge amount of information generated. Crowd control is one of the most important challenges that today's video systems face. From a computational point of view the first problem is how to conceptualize "crowd". Saxena *et al.* [2008] defines crowd as a region corresponding to more than one person which has coherent and homogeneous motion. A natural approach would then be to detect people, determine if there is more than one, and analyze its motion. Such an approach would belong to the so called "object-based methods". Junior *et al.* [2010] divides the literature about crowd analysis into "object-based" and "holistic" methods.

Object based methods aim to analyze the group behavior through its individual components, while holistic ones look at the crowd as a global entity. We can easily associate the former with high level local features and the latter with global ones. Detecting and counting people in a crowd might not be easy due to segmentation problems. Tracking people might also be a difficult task in a crowd, even for humans. In fact when a person sees a crowd it is likely that he has not counted people, and nevertheless can affirm there is a crowd. Another important issue is to define the concept of "region". Watching ten people on a street, if there are enough space between them, would not be considered as a crowd by many. An object-based approach would have to add constraints about the distances between objects.

On the other hand an holistic approach would rather represent the "impression" (3.1.1.3) of a crowd. Instead of trying to segment people, which in crowds is difficult (Marques & Llach [1998], Tu *et al.* [2008]), we could just treat it as an object and avoid the logical division in persons, which is only useful if we actually need to count people. Junior *et al.* [2010] suggests that holistic methods present

better results in very high-density crowds, which make sense, since segmentation is harder in these environments.

Following these ideas, we wish to test if human activity can be addressed by means of global features, such that logical division can be avoided.

4.1.1 Method

Since we are dealing with video and search for activity, background subtraction seems to be a good choice to extract the relevant part of the image where action takes place. We select simple global features to represent shape (Teague [1980]), color (Almrabet *et al.* [2009]) and texture (Haralick *et al.* [1973]). These features are calculated for the segmented part of each frame of a video sequence and put together in a feature vector (FV), one per frame. Then a supervised learning method is trained using frames labeled with the different crowd behavior.

Each FV is labeled with label "1" if it has been generated from a frame that belongs to event E_n or "0" if it does not. We build up a training set with FV labeled "1" or "0". Once the MLP is trained its output will be used for labeling testing FV, thus determining if it belongs to an event or not. This way we define a two-class classifier specialized in one kind of event. If more than one event are to be recognized we would need to train one MLP for each event following the same procedure. A frame can be recognized by two different MLP, this could have different meanings. It can be used to code new events, detect transitions etc. Further logic may be applied with MLP outputs in order to define complex behaviors.

4.1.2 Experiments

4.1.2.1 Data preparation

We test our method with PETS 2009. The dataset provides footage from a multi-camera installation. Set number 3 (event recognition) has four different cameras in four different positions recording at the same time. A camera position is called view, therefore we will have four different views. We will only use set number 3 since it is the one designed for event recognition testing. Dataset 3, event

recognition, is divided in four video sequences identified with four time stamps (14-16, 14-27, 14-31, 14-33). Each view has the same sequences, so we have 16 video sequences to work with. Sequences are recorded at 7 FPS lasting between 19 and 58 seconds.

In order to increase the number of frames available for each experiment we put together all the frames from each view in four new folders. We discard mixing frames from different views because the extracted features make no sense when changing from one camera view to the other. Another important issue is that depending on the view, the starting and ending frames of an event may change.

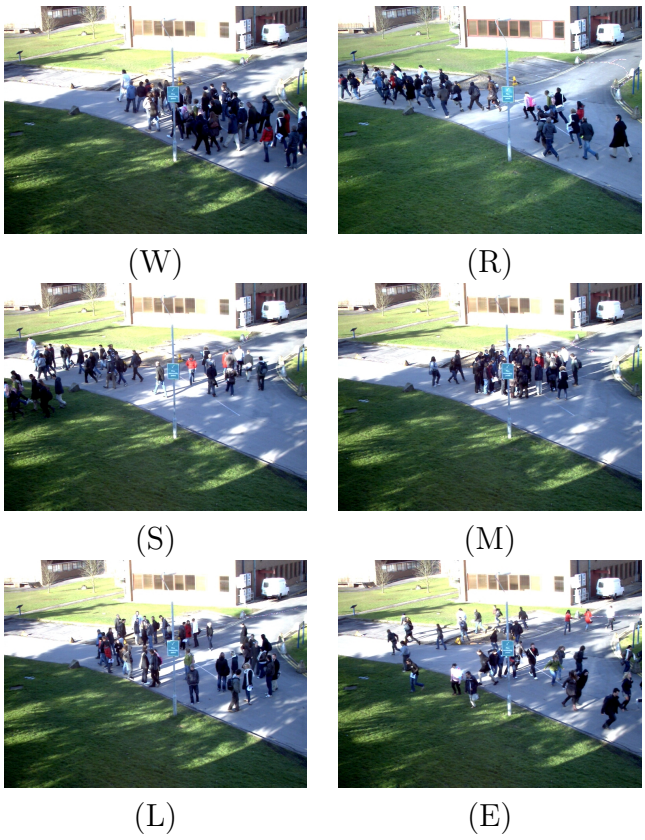


Figure 4.1: Events: (W) Walking, (R) Running, (S) Splitting, (M) Merging, (L) Local Dispersion, (E) Evacuation

4.1.2.2 Training

Each frame is labeled with "1" if it contains an event E_n or "0" if it does not. So it is important to state when an event starts or ends. We will try to recognize six events: walking (W), running (R), local dispersion (L), splitting (S), merging (M) and evacuation (E). One frame may contain more than one event. Table C.1 shows the video script containing event information for every view. This script has been done by evaluator number 1. We asked three other people to fill in script for View 1 in order to evaluate the differences between people classifying the video footage using the same event definitions provided by PETS organization, results are shown in Table C.2.

We build up six training sets, one per kind of event (W, R, L, S, M, E) to get one MLP for each event and each view. To create a training set we use the same proportions that before, we select randomly 75% of the frames that belong to event E_n for training and 25% for testing. Training and testing sets are completed with frames that does not contain event E_n, with a final distribution of approximately 40% FV with the event to be learned and 60% without it.

4.1.3 Results

Table 4.1 shows errors for the six events for each view using script by evaluator 1. In all the cases more than 88% of the FV are classified correctly, in most cases more than 95%. The "worst" view is View 2. This is likely to happen because it is harder to make a difference of people running, walking or merging from a frontal point of view. The hardest event to be recognized is Local Dispersion.

	View 1	View 2	View 3	View 4
Walking	1.13%	4.13%	1.50%	2.63%
Running	2.38%	4.06%	3.25%	1.67%
Splitting	1.23%	3.70%	5.95%	2.47%
Merging	0.75%	6.01%	0.38%	2.64%
Dispersion	7.84%	11.76%	7.02%	10%
Evacuation	0%	4.17%	0%	0%

Table 4.1: Event errors per view

Table 4.2 shows errors for the six events using scripts from all the evaluators

but only for View 1. The error rates from evaluators 2, 3 or 4 are very similar to the ones obtained training with evaluator 1 script, most of the cases show more than 95% of FV properly classified. This is by far much better that the error rates we see in Table C.2, when we compare the classification done by evaluator 1 and the others, just a few cases have error rates below 95%.

	Evaluator 1	Evaluator 2	Evaluator 3	Evaluator 4
Walking	1.12%	1.12%	2.53%	0.87%
Running	2.38%	1.67%	0%	3.10%
Splitting	1.23%	2.30%	4.17%	1.07%
Merging	0.75%	2.63%	1.50%	1.89%
Dispersion	7.84%	6.06%	3.79%	4.76%
Evacuation	0%	0%	0%	0%

Table 4.2: Event errors per evaluator

Finally, Table 4.3 compares the results for evacuation generated using two different algorithms. The first one is the result we get using MLP as seen above. But the second one is the result of the combination of two MLPs, splitting and running MLPs, since evacuation can be seen as the co-occurrence of both. To perform this test, we did classify all the frames from the different Views using the trained MLPs. Then we looked at all the frames with running and splitting event and compared them with evaluator 1 script to see if it was a correct classification or not. This test has no previous training to define evacuation.

	MLP	AND
View 1	0%	0.37%
View 2	4.17%	0.56%
View 3	0%	0.37%
View 4	0%	0.19%

Table 4.3: Comparation of methods for evacuation

4.1.4 Discussion

Four people given the same instructions label frames differently, which confirms that conceptualizing crowd behavior is not simple. In fact complex events, like

"dispersion" or "evacuation" present higher differences. However, our method based on global features achieves very low error rates, without need to segment or track people.

The "gist" or "impression" of the region where action takes place is enough to differentiate between several crowd behaviors. We have used a simple method for segmentation, background subtraction, which is an efficient method to detect activity. Then we have trained one recognizer per event category. These categories can also be seen as high level features that might be comprehended to find another category. For example, the concept "evacuation" might be conceptualized directly with a specialized recognizer, and or indirectly by the logic combination of the results of two recognition modules, "splitting" and "running". In three of the four views the direct approach has slightly better results, however the mean error in the indirect is a third of the direct approach, 0,3725% versus 1,0425%.

The method is extremely accurate, and in fact the errors are questionable. For example some frames labeled as "local dispersion" can easily be considered as "merging" frames. This kind of ambiguity is natural and can easily be resolved with the context generated by the following frames, with more information the system is able to increase its reliability.

4.1.5 Conclusions

Different activities can be successfully categorized by a set of low level features. Even when the activity is defined by the actions of a set of people, the activity can be recognized with global features. The whole can be categorized without categorizing the parts. The only segmentation needed is the one that divides background and foreground objects. This segmentation is simpler and avoids the generation of segments and their comprehension. Learning is achieved with examples, without need of rules describing the behavior of each element. The activity is recognized as a global relation between all the elements that have moved in the image.

4.2 Authorship perception

The second example is about writer identification for police investigations and is based on Cermeño *et al.* [2014a]. Handwriting can be considered as a biometric feature, and thus can be used to determine the authorship of a document (Srihari *et al.* [2002]). Tapiador & Sigüenza [2004] describes a manual method used by police experts to compare different questioned documents that consist on the classification of relevant characters based on their shapes. Each document is then formulated using the results of the classification of its characters. We could express this process in the terms of perception: in order to perceive the authorship of a document, high level features, the characters, are classified to generate a representation of the document. This example is interesting, because unlike most of the examples of perception we could quickly think of (e.g.:object recognition), authorship perception is not immediate, it requires some effort and an explicit methodology.

One could question if writer identification is actually perception. Under our definition "picking up and categorizing information related to an image" the answer is yes. Aside from our definition, if we consider that face recognition is an act of perception, why shouldn't it be handwriting authorship recognition? Both are visual biometrics. In fact, sometimes face recognition is not immediate and requires the recognition of parts, for example explicit eyes or nose description.

Tapiador & Sigüenza [2004] also proposes a computer based method inspired in the manual one to speed up and increase the reliability of the process. Following the steps that a policeman would undertake, each character is considered individually, segmented as a new image and labeled with the letter it represents. This process is done with the support of a digital image-manipulating tool, and therefore requires human intervention to properly segment and categorize. Then letter representations from an questioned document (unknown author) are automatically compared to the representations of the same letters from documents from known authors. This approach can be divided in two cycles: the first one would segment and categorize each character, and the second one, would integrate the information from the first one to categorize the image and conclude who is the author of the handwriting.

English handwritten characters classification can be done by a machine achieving an accuracy over 99% (Ciresan *et al.* [2011]). However segmenting handwriting into characters has proved to be more difficult (Stamatopoulos *et al.* [2013] Saba *et al.* [2011]). Lu & Shridhar [1996] claims that "it is extremely difficult to segment characters in handwritten words without the support from recognition algorithms. Therefore, unlike the problem of machine printed character recognition, the handwritten character segmentation and recognition are often closely coupled". The big difference between machine printed and western handwritten character segmentation is that the former are always separated by spaces while the latter may have some of the characters of a same word touching. The constraint used to segment machine printed words is not reliable for handwritten ones.

On the other hand, finding touching words in a sentence is unlikely, and therefore space separation can be much more reliable for word segmentation than for character segmentation. Instead of following a bottom-up approach, picking up each character of a word to recognize it, we can try a top-down one where a direct hypothesis is made about the whole word before categorizing each of its parts, in such a way that segmentation is guided by information extracted from the whole. For example Koerich *et al.* [2005] proposes to combine two different classification strategies operating in different representation spaces (word and character) in order to improve word recognition. While Larson [2004] discusses if humans use word shape or letters to recognize a word, Rehman & Saba [2012] states that "Regarding word recognition, the problem is seemed to be solved in small and static lexicons using holistic strategy. However, recognition accuracy dropped significantly for larger lexicons. Therefore, segmentation based word recognition is an alternative solution".

Whereas handwriting variability is a problem for word recognition, it is a strength for writer recognition. Without variability it would be impossible to distinguish one author from the other. Works like Tapiador & Sigüenza [2004] exploit the variability in the writing of letters, but as we have seen they face the character segmentation problem, which derives from the variability in the way an author joins letters. Thus, why not using this variability to perceive the authorship of a document?

Joint letters are connected pieces of ink. In handwriting analysis we usually call them connected components. A writer could be considered as a stochastic generator of connected components (Schomaker *et al.* [2004]). Depending on the writer, these components may codify a character fragment, a character, a word fragment or even a complete word. The segmentation procedure extracts connected components from a scanned document. Schomaker *et al.* [2004] suggests to generate a codebook with connected components, and thus replace the classification of characters with a classification of connected components, which are easy to segment. This kind of methods simplify the process of segmentation and rely on the recognition of high level features, the connected components (COCOs).

Connected components can be described with different kinds of features. Grapheme-based features describe COCOs by mapping their local structures into a common space. Schomaker & Bulacu [2004] and Schomaker [2008] are two examples of methods using grapheme-based features. More recently Christlein *et al.* [2015] describes COCOs using Zernike moments extracted on contours and encoded into a Vector of Locally Aggregated Descriptors (VLAD). Xiong *et al.* [2015] follows the bag-of-words model with SIFT features.

On the other hand we have texture-based features that consider handwriting as a texture, such that writer identification becomes a problem of texture recognition. Local binary patterns (LBP) and Local phase quantization (LPQ) generate useful descriptions, such that COCOs can be compared (Hannad *et al.* [2016]). Some texture-based approaches avoid COCOs evaluation and directly compare handwriting textures with promising results (Bertolini *et al.* [2013]; Nicolaou *et al.* [2015]).

Segmentation is sometimes seen as a problem for writer identification, like for many other computer vision applications. However experiments have shown that COCOs produced by a simple segmentation algorithm can be analyzed in terms of probability distributions that are able to characterize upper-case handwriting (Schomaker & Bulacu [2004]). To deal with more realistic situations, where upper-case and cursive handwriting coexist, Schomaker *et al.* [2004] extends the previous work with the introduction of "fraglets", which are segments of COCOs. The hypothesis behind segmenting COCOs is that working at word or syllable level may be confounding because it makes writer identification depend on the

content of the text. Other publications using fragmented COCOs consider that fragments have a higher discriminative power than COCOs (Bensefia *et al.* [2005]; Hannad *et al.* [2016]).

We do not try to segment handwriting into characters, which has proved to be difficult, but neither do we try to segment COCOs into fragments. We focus on simple segmentation of handwriting into COCOs and evaluate the effects of relaxing the fundamental constraint that defines a COCO: connection. A connected component is represented by a set of foreground pixels, in which any pixel has at least one neighbor member of the set. We propose a new framework for segmentation based on different definitions for the concept "neighbor". The constraint "neighborhood" is enlarged from touching elements to elements closer than a determined distance. Multiple distances are used to generate COCOs, so that we call it multi-segmentation. The hypothesis is that juxtaposition may not be the best constraint to define neighborhood, and that exploring more than one segmentation space might generate useful information for writer identification. In other fields, like object recognition, adopting different segmentation strategies has proved to be effective (Uijlings *et al.* [2013]).

4.2.1 Method

4.2.1.1 Handwriting segmentation

The first idea of our method is that when we segment an image we are creating a relation between the segments that represent information valid to identify the authorship of a document. The second idea is that using more than one parameter for segmentation might be useful. Resolving power, the ability to distinguish two adjacent pixels is a basic way of segmenting an image. When we are not able to group pixels together, for example when zooming in a high resolution picture, perceiving the whole becomes harder. If we relax the constraints to consider that pixels belong to the same segment, perceiving wholes is easier but we sacrifice the perception of details. In this case we are not interested in details but in analyzing the image as a whole, so we can relax the constraints of segmentation. To do so, we consider that two pieces of ink may be connected even if there are some background pixels between them. Actually what we do is to simulate that

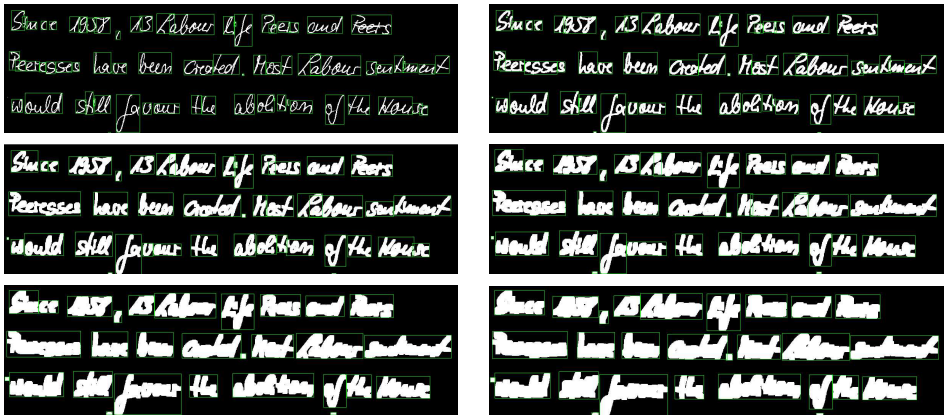


Figure 4.2: Growth levels: 0,2, 4, 6, 8, 10

a thicker pen goes over the handwriting in what we call a process of "growing the connected components". Instead of having just one pen, we simulate several pens of different sizes. Each pen represents a different segmentation, leading to a different set of COCOs, and hence the name multi-segmentation for our method.

It is composed of the following steps. We first binarize the image in order to get rid of noise coming from the original document, its scanning process or image compression. We complement the image, such that background pixels become black and foreground pixels white. Then we apply a simple segmentation algorithm. We scan the image line after line, labeling each pixel. For every white pixel a new label is created only if there are no neighbors already labeled. If there is one its label is adopted. In case of several neighbors the first label is adopted and expanded to the other neighbors. At the end each group of pixels with the same label is considered as a connected component. The rectangle containing a connected component is called block.

In order to implement a multi-segmentation scheme we need to apply a growth algorithm to the image. Every white pixel neighbor becomes white. The growth algorithm admits a parameter to fix the growth radio. The segmentation algorithm is then applied over the grown components.

Depending on the image resolution and the writers characteristics different growths are required to compose letters, then syllables and then words.

4.2.1.2 Size analysis

Our first approach tries to describe an author's handwriting by using the number of connected components generated. We take advantage of what we presented as a problem: how a writer groups the pieces of ink. If a COCO corresponds to a letter, a syllable or word is not relevant. We focus on how many COCOs the writer generates and their size.

In order to compute this we calculate:

- Histogram of block size
- Histogram of block ratios (width to height)

Block size could be a problem in some cases, for example if an author uses bigger letters than usual. However police experts expressed their interest on these features since very often one person writes within the same size range. Block ratios are not affected if an author changes the size of its writing. Empirical tests show that 8 to 12 histogram bins provide optimal results.

Multi-segmentation brings the chance to multiply the number of features and make them robust against an author's writing size change from one document to another. For each growth we calculate the histograms described above. Then for each histogram bin we calculate its evolution ratio, this is, how much it has increased or decreased from the previous growth level. With all these data we can build a feature vector for every writing sample. This feature vector is a compilation of several probability distributions related to the whole image. When two samples are compared, the images as wholes are compared, not their parts.

In order to identify an author we need a classification algorithm. Two classifiers are considered for writer identification, Euclidean distance (ED) and Multilayer Perceptron (MLP). Schomaker & Bulacu [2004] suggests the use of MLP or Support Vector Machines (SVM) for writer verification but has some objection to apply them for writer identification. Gazzah & Amara [2008] proposes to use ensembles of MLP to classify features extracted from Arabic handwriting, as well as trained SVM. MLP provides slightly better results than SVM (94,7% vs 93,76%) classifying structural and global features. For this work we implement an ensemble of MLP, one per writer, with bipolar output indicating if the vector

belongs to the selected writer or not. The output is also considered when several MLP recognize positively the same vector.

4.2.1.3 Shape analysis

Size is just one type of feature among many. Our second approach extracts features related to the shape of COCOs. Like in the previous section we wish to represent a sample of handwriting by a probability distribution of its COCOs components, but this time grouped by shape features. For each writing sample we generate COCOs using the multi-segmentation method, then we extract the shape features presented by Perez *et al.* [2014]:

- Mass: the number of pixels in the image that contain handwritten text. This measure is normalized to the size of each image and it is in the range [0,1].
- Center of mass: the unique point where the weighted relative position of the distributed mass sums to zero. This measure is normalized to the size of each image and is in the range [0,1].
- Eccentricity: the ratio of the distance between the foci and the major axis length of the ellipse that has the same second-moments as the object. The value is between 0 and 1.
- Orientation : the angle in degrees between the x-axis and the major axis of the ellipse that has the same second-moments as the object. The value is in the range [-90, 90].
- Euler number : scalar that specifies the number of strokes in the text minus the number of holes in the text. A hole is defined as a space inside a stroke.
- Solidity: the ratio of the mass of the text and the area of the convex hull of the text. The convex hull is defined as the minimum convex perimeter that can contain the text.
- Extent: the ratio of the mass of the text and the area of the bounding box that contains the text. The bounding box is the smallest rectangle containing the text.

Each COCO has associated a Connected Component Feature Vector (COCOFV). The method to create the probability distribution is similar to the one described by Schomaker *et al.* [2004] or Schomaker & Bulacu [2004] but without contours. A a self-organizing map (SOM) (Kohonen [1990]) is used to create a codebook with COCOFV. In this work the SOM consists of a matrix of nodes. Associated with each node is a weight vector of the same dimension of the input, in this case a COCOFV. All the weight vectors are initialized with random values. Then COCOFVs from writing samples selected for training are compared with each node weight vector. The closest node is updated using a learning rate of 0.1, so the node weight vector becomes more similar to the FV. We repeat this learning algorithm for 500 epochs with all the COCOFVs from the training set.

Every writing sample could be described as a probability distribution of the COCOFVs in the SOM. The classification or mapping phase consists in calculating the number of COCOs similar to each node of the SOM. Thus we have a new vector called writer feature vector (WFV) with as many components as nodes in the SOM. Each component has the probability that the COCOs from the sample are similar to its associated node. A writing sample is represented by a WFV.

In order to compare known author writings with questioned samples (unknown authors), we compute the WFV of the questioned document and compare it with the WFV of samples from known authors. We assume that the author of the questioned sample is the one that wrote the "known sample" which WFV is closer to the questioned sample WFV. We use the Euclidean distance to compare the WFV. Growth 1 COCOs are the ones from growth 0 plus the ones from growth 1, growth 4 COCOs include COCOs from all the previous levels, 0,1 ,2 3 and also 4. The higher the growth level the more COCOs to describe the sample. When two samples are compared, the description of each image as a whole is compared, not its parts.

4.2.2 Experiments with a small group of authors

4.2.2.1 Database description

The first database used in these experiments was created with spanish police corps using writing samples from real investigations. Samples contain mixed-

style handwriting with upper-case and cursive styles from 8 different authors. The documents have been "treated" by forensic experts and contain arrows, circles and other symbols used for manual handwriting recognition that may add noise to the automatic recognition. Scanning is done with a resolution of 300 DPI, after that the only manual processing on our side is to cut a region of interest containing between 50 to 80 words. This is done by cropping a rectangle from a document scanned using a basic image edition tool. No special care has been taken so samples may have non aligned text or truncated words in the edges.

We build three testing sets using documents from the database. For the first one TS1 we split 8 documents from different writers into 16 samples. One sample will be considered unknown and the second one known for each author. The second set TS2 is composed of 10 samples, 2 from each writer. In this case we use different documents. The third set TS3 is done using TS1, and composed of 16 samples in each group, "known" and "unknown". We split each known sample in two new samples in order to have smaller "known" samples. Then we join the original two groups of samples to form the unknown samples. The idea of this set is to test with small samples (less than 40 words).

4.2.2.2 Feature vector generation

We decide to divide the feature extraction in three phases A, B and C. Phase A includes growth radios of values 2, 3 and 4 pixels. Phase B growth radios of values 6, 8 and 10. Phase C includes growth radios 10, 12 and 14. Segmentation is processed over the samples after applying the different growths in order to obtain the connected components.

For each block containing a connected component we calculate the size and width-height ratio in order to obtain a probability distribution for a given growth and writer. This distribution is coded in two histograms with 10 bins each. Since each phase is composed of three growth steps we generate 60 features. After that the feature vector of each phase is completed with the ratios of equivalent histogram bins within the phase, adding 10 new features per histogram and evolution ratio, thus 40 in total (one phase has two ratios calculated between three growth steps). In our example with three phases the size of one writer's feature

vector is 300. Vectors from known writers would be used for training whereas unknown samples are used to test the system.

4.2.3 Results for a small dataset

The following tables contain the results from the classifiers. When using MLP, after executing all the tests, "unknown" and "known" groups are swapped so the training files become testing files and testing files become training files. Results show the average.

	0	A	A + B	A + B + C
TS1	37.5%	43.8%	18.8%	12.5%
TS2	50%	60%	30%	30%
TS3	15.6%	15.6%	9.4%	3.1%
Mean	34.4%	39.8%	19.4%	15.2%

Table 4.4: Error rates with MLP classifier

Table 4.4 presents the results of MLP using features from one, two and three phases over the different testing sets. We can see that for all the sets performance is improved when more phases are added. TS1 with all the growth phases has an error of 12,5%, this is one mistaken authorship over eight. Without multi-segmentation the error rate is 3 times higher (37,5%), while random classification would show 87.5% of error. TS3 shows even better results with zero or one error depending on the training group selected, thus an error rate of 3%. The worst results are found in TS2, with one or two mistakes over five writers.

	0	A	A + B	A + B + C
TS1	12.5%	62.5%	25%	25%
TS2	60%	60%	20%	40%
TS3	31.3%	56.3%	50%	43.8%
Mean	34.6%	59.6%	31.7%	36.3%

Table 4.5: Error rates with ED classifier

Table 4.5 presents the results using ED. The improvements derived from the use of multi-segmentation are not very clear. Most of the results are worse than

the ones obtained without growth phases. But in all the testing sets results are improved when adding phases B and C to A.

	M1	M2	M3	M4	M5	M6	M7	M8
W1	4	0	0	0	0	0	0	1
W2	0	4	0	0	0	0	0	0
W3	0	0	4	0	0	0	0	0
W4	0	0	0	4	0	0	0	0
W5	0	0	0	0	4	0	0	0
W6	0	0	0	0	0	4	0	0
W7	0	0	0	0	0	0	4	0
W8	0	0	0	0	0	0	0	3

Table 4.6: Confusion matrix for TS3 with actual Writer classes Vs predicted MLP classes

We observe that MLP results are better than ED when using more growth phases, without multi-segmentation there is not a clear improvement between MLP and ED.

Table 4.6 shows the confusion matrix for TS3. As stated before we have two samples per writer in the training set and two different ones on the testing set. Once we have finished the classifications with these sets, we swap them to classify again (training becomes testing and testing becomes training). Thus for each writer 4 samples are classified. Just one error is reported, one of the samples from writer 8 is considered to be from writer 1.

4.2.4 Experiments with a large group of authors

4.2.4.1 Database description

In order to evaluate multi-segmentation with a larger dataset we have selected IAM dataset (Marti & Bunke [2002]). This database contains forms of different handwritten English texts which were scanned at a resolution of 300dpi and saved as PNG images with 256 gray levels. More than 650 authors have contributed to the dataset but most of them with just one page, many others have only a couple of lines. For our experiments we have built a set with 100 writers each of them with two samples and more than two lines, a total of 200 samples.

Like in the previous dataset, the only manual processing is to cut a rectangular region with the handwriting inside. IAM samples do not have forensic annotations and the writing is of better quality. With quality we mean that it is clear that authors of IAM dataset took some time to write the samples and were aware that others would use them. In the dataset from the police corps, authors wrote short texts with instructions, orders or reports that were very likely to have just one receiver and were going to be used once, so no special care was taken with the calligraphy.

With basic segmentation (no growth) the number of COCOs generated is most of the times between 200 and 300. If we apply the same growths than in the previous experiments (0,1,2,3,4,6,8,10,12,14) the number of COCOs per sample is usually around 1500. So with just 100 samples we were able to generate 150K COCOs which is similar to the 152K COCOs used by Schomaker *et al.* [2004] to train the SOM, but we required 100 pages instead of 300.

4.2.4.2 Baseline

IAM is a public dataset so we can use it to compare and evaluate the effects of multi-segmentation with state-of-the-art techniques. Local Binary Patterns (LBP) and Local Phase Quantization (LPQ) are two texture descriptors that have shown some of the best results in the IAM dataset (Bertolini *et al.* [2013]; Hannad *et al.* [2016]). We implement both techniques to describe each connected component.

Connected components are created using the same algorithm as the one in the previous experiment but without growth. Each document is represented by a set of COCOs. Each COCO is described by a texture descriptor. We first test with LBP and then LPQ, the description given by these methods is an histogram. We have used the same dissimilarity measure than Hannad *et al.* [2016] but with euclidean distances. The dissimilarity between a document K from which the author is known, and a query document Q from which the author is unknown is:

$$DIS(Q,K) = \frac{1}{card(Q)} \sum_{i=1}^{card(Q)} \min_{h_j \in K} (distance(q_i, h_j)) \tag{4.1}$$

where q_i and h_j are the histograms generated by LBP or LPQ from COCOs i and j from a pair of documents Q and K. The distance between histograms is computed as follows:

$$distance(q_i, h_j) = \sum_{n=1}^{NDim} \left| q_{in} - h_{jn} \right| \tag{4.2}$$

where $NDim$ is the number of bins of the histogram. For LBP we have used 59 bins and a radio of 8, while for LPQ we have used 256 bins and a radio of 1.

The writer identified for document D is the one which wrote the document K, whose dissimilarity is lower:

$$Writer(Q) = arg \min_{K_i \in RefBase} (DIS(Q, K_i)) \tag{4.3}$$

We use the concept of Top-N to refer to the n best candidates to be the author of a writing. The best candidates are the set of n candidates whose dissimilarity $DIS(Q, K_i)$ is lower. When the author of a questioned document is in the set of candidates we consider it as a successful classification.

4.2.4.3 Multi-segmentation and local descriptions

In order to test the effect of multi-segmentation we compare the results of the baseline study with the ones given by a combination of features generated by multi-segmentation with the local descriptors (LBP and LPQ) used in the baseline. For these experiments we have chosen the features based on shapes. Each questioned document is represented by the distribution of its COCOs in a SOM, the Writer Feature Vector (WFV). We can therefore calculate a dissimilarity between a questioned document Q and a document from a known author K using the following formula:

$$DIS(Q, K) = \sum_{n=1}^{NDim} \left| q_{in} - h_{jn} \right| \tag{4.4}$$

where $NDim$ is the number of bins of the histogram, which in this case is the number of nodes of the SOM. In the experiments with SOMs presented in Schomaker *et al.* [2004], results improve from 5x5 to 20x20 an then remain stable.

We have used a 20x20 SOM, and thus histograms have 400 bins. q_i and h_j are the histograms generated for questioned document i and document j for which the author is known. This approach represents a document with one histogram, whereas the local approach represents a document with a set of histograms, one per connected component.

We combine both by adding the result of 4.1 and the results of 4.4, such that we have only one dissimilarity measure for each pair (Q,K). Then we can identify the writer using the formula 4.3. The evaluation is also done with the measures Top-N.

4.2.5 Results for a large dataset

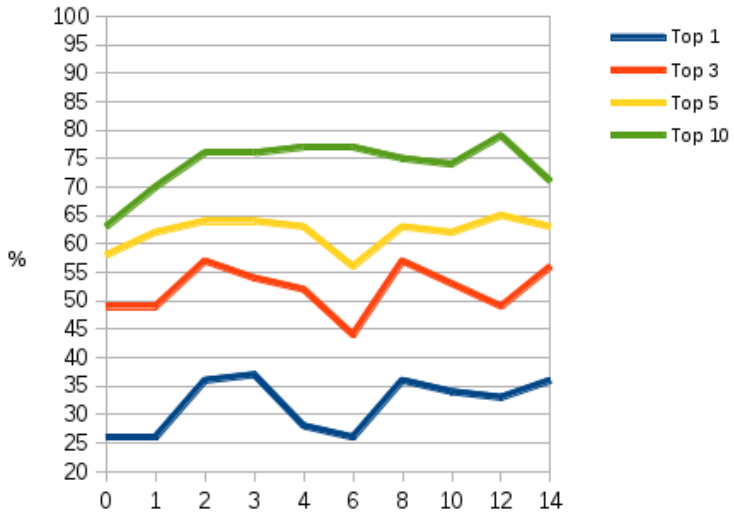


Figure 4.3: % of authors successfully identified using only a SOM. The number in X axis represents the higher growth level from which COCOs are included.

Figure 4.3 shows the evolution of the percentage of documents successfully classified using only a global description histogram based on the SOM for different growth levels. The best result for Top-1 is 37 % with COCOs from growths up to 3. The best result for Top-10 is 79 % with COCOs from growths up to 14. Using results from multi-segmentation improves the results of simple segmentation (growth 0) in all the cases except one (growth level 6).

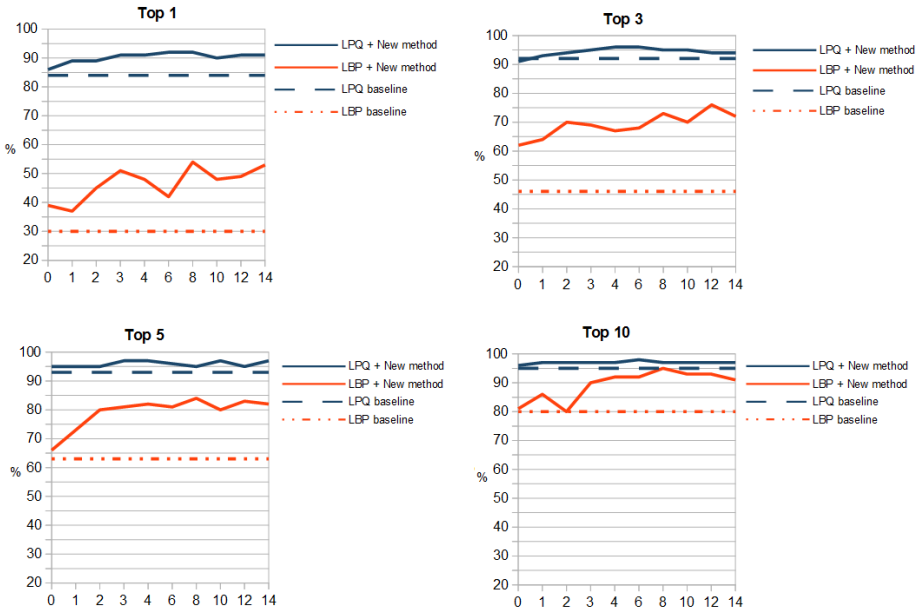


Figure 4.4: % of authors successfully identified using SOM and local descriptors (LBP & LPQ). The number in X axis represents the higher growth level from which COCOs is included

All the results for the different Top-N have been tabulated in C.3. In all them except Top-3, we observe that growth levels 2,3 and 12, 14 present similar values. Successful classifications increase from no growth to growth levels 2 and 3, but then fluctuate without a clear tendency. The worst results are given by growth level 6.

Figure 4.4 presents the evolution of the percentage of documents successfully classified using a dissimilarity that combines both a global and a local description for different growth levels and different top-n measures. The best result for Top-1 is 92% with COCOs from growths up to 6 or 8 and LPQ. The best result for Top-10 is 98 % with COCOs from growth 6 and LPQ.

The results using LPQ as local descriptor are high in all the cases. We observe some improvements when the first growths are included but then results fluctuate in a band of 3%. On the other hand the results from local descriptors implemented with LBP increase when features from more growths are added. For example Top-1 shows an improvement of 76.6%, from 30% to 53%. All the results for Top-1,

Top-3, Top-5 and Top-10 are detailed in the appendix C.

4.2.6 Discussion

4.2.6.1 Multi-segmentation for a small group of writers

When analysing multi-segmentation on its own we reach 97% success rate in writer identification with a text-independent database of 16 real short samples from 8 different authors. These results are similar to the ones referred before Said *et al.* [1998] with 10 writers. The original documents have noise, upper-case or cursive styles. The process is full automatic once a piece of paper with more than 20 words has been digitalized without need of any customized pre-processing like the one required by Said *et al.* [1998].

Comparing results of basic segmentation (no growth) and the different growth phases within multi-segmentation suggests that writers characteristics may not always be in the same level of segmentation. With basic segmentation there is little difference between simple ED and MLP, almost 34% error rate. When we use MLP results become better and better when we add more phases, whereas ED average error rate keeps more or less the same. MLP is able to assign different weights to the features from each writer while ED is not, so MLP can give more importance to the features coming from relevant segmentation levels. In these results we find somehow the idea that a unique partitioning of objects should be rejected in favor of a combination of multiple partitioning strategies combined (3.3.1.2).

4.2.6.2 Multi-segmentation for a large group of writers

Table 4.7 compares the best results for datasets with 100 writers from Bertolini *et al.* [2013] using LBP and LPQ, from our baseline study using the same local descriptors, LBP and LPQ, from writer identification based on multi-segmentation SOM descriptors and from writer identification based on a combination of global (multi-segmentation SOM) and local (LBP or LPQ) descriptors.

These results suggest that multi-segmentation descriptions are not good enough to identify writers when the number of authors is increased. The effect of adding

Bertolini *et al.* [2013] LBP	51.7%
Bertolini *et al.* [2013] LPQ	88%
LBP	30%
LPQ	84%
Multi-segmentation SOM	37%
LBP & Multi-Segmentation SOM	54%
LPQ & Multi-Segmentation SOM	92%

Table 4.7: Comparison of Top-1 measures for a dataset with 100 writers

more segmentation growth levels (figure 4.3) does not show any clear improvement beyond the first or second increment.

Nevertheless figure 4.4 shows how the combination of multi-segmentation descriptors and local descriptors has a positive effect. We have implemented LBP and LPQ descriptors as baseline study. In the first case, adding multi-segmentation descriptors increases the success rate in all the cases, up to 65.2%, from 46% to 76%. LPQ baseline is also improved, but in a lower measure, the best improvement is 9.5 % from 84% to 92%. This result reaches the ones presented by state-of-the-art methods. It is 4% higher than Bertolini *et al.* [2013] for a dataset with 100 writers, and 4.7% lower than its best result for 650 writers.

The results of figure 4.4 suggest that adding more growth levels increases the accuracy of the identification process up to a determined level, in which it stabilizes, or even decreases. We may think that too big COCOs are less characteristic than smaller ones. The results of growth 6 are specially interesting, while classifying only with multi-segmentation descriptors shows the worst percentages for growth 6, when combined with LPQ, they show the highest ones for Top-1, Top-3 and Top-10. We do not see any direct relation between the accuracy of multi-segmentation descriptors used alone or in combination with local descriptors.

4.2.7 Conclusions

Instead of trying to emulate how a human expert would segment handwriting , which is usually a difficult task (Tapiador & Sigüenza [2004]), we process con-

nected components. Segmenting handwriting into COCOs is simpler than segmenting into characters because the constraints that must be satisfied are simpler. The same reasoning applies to COCOs and their fragments. Segmenting into fragments requires an extra operation with its own constraints. The only constraint that determines whether a pixel is connected to another or not is the distance, the number of background pixels, between them. Instead of adding more constraints our method segments the same image with multiple distances, which in our experiments are represented by "growth levels".

Descriptors based on multi-segmentation outperform descriptors based on normal segmentation. For a small group of writers, a system can learn how to differentiate writers only by analyzing the evolution of the probability distribution of COCOs over the number of pixels defining neighborhood. We had no new of categorizing each COCO. However for a large group of writers, this information is not enough. We have shown that a combination of COCOs descriptors, which are local features of the handwriting sample, with COCOs probability distribution, which is a global feature, achieves state-of-the-art results.

Using multi-segmentation to generate the COCOs probability distribution instead of simple segmentation increases the accuracy of classification in all the cases. Under the multi-segmentation framework, improving writer identification becomes a parameter estimation problem, in which we must find the best distance (growth level) to generate COCOs. We have seen how the best growth level changes from one experiment to the other and therefore we have demonstrated that searching more than one level gives better results. This case is an example of why the process *Divide* should be included in a loop like the one shown in section 3.2.2.4. Exploring the space of possible relations between foreground objects improves the accuracy of perception.

4.3 Intruder perception

The third example is about intruder detection for perimeter protection using video surveillance systems and is based on Cermeño *et al.* [2017b]. An intruder is something or someone that should not be there, in an image it is represented by one of its parts. The challenge is therefore to find how to characterize this

part, how to segment it. In section 3.3.1 we classify segmentation methods into sliding window, semantic and motion detection. An intruder can be a person, so one could think of person detection methods. Ogale [2006] divides techniques for human detection from video in two groups, the first one requires background subtraction, whereas the second one does not. The latter are called "direct detection", assuming that background subtraction is a pre-processing phase. The problem with the so called "direct detection" techniques is that intruders may not have human appearance, and thus avoid the recognition. For perimeter protection assuming that intruders have the appearance of a person is not secure, not only because of poor quality in the video but also because people can influence it. They can crawl, creep or wear costumes to change the way they look.

On the other hand, background subtraction is a motion detection algorithm independent from recognition. In perimeter protection, considering motion as a characteristic of the intruder has many advantages. If the cameras are properly installed, an intruder must move on the protected area before becoming an intruder, so static elements are more a potential intruder than a real one. Motion detection is independent of the form of the intruder. Background subtraction techniques usually implement some kind of background model, which actively learns changes in the environment, so that the perceptual system can adapt. Without adaptation, the same detection would be reported again and again. These techniques have a low complexity, most of them between $O(1)$ and $O(5)$ (Piccardi [2004]). Such a complexity is impossible for exhaustive search algorithms like sliding window, unless we reduce the number of scales and overlapping allowed for the windows.

Background subtraction has also limitations. Brutzer *et al.* [2011] lists the main challenges for background subtraction techniques: gradual illumination changes, sudden illumination changes, dynamic background, camouflage, shadows, bootstrapping and video noise. Camouflage is just a resolution problem, which can be found in any segmentation algorithm. For perimeter protection bootstrapping is not required, since it tries to solve the problem of creating a background model from initialization data with foreground objects, which is unlikely for a protected area. Except some types of shadow (Sanin *et al.* [2012]), the other challenges are usually handled with improvements in the background

modeling (Bouwmans [2014]).

In this section we build an intruder detection system with state-of-the-art techniques, in order to test it with thousands of hours of video from cameras used to protect perimeters of real sites. The experiments are divided in two. The first experiments aim to show the performance of the system in normal situations, how reliable its detection is and how well it adapts to the natural changes in illumination. The second experiments deal with abnormal situations, in this case sudden illumination changes. Instead of proposing a new background modeling algorithm or people detector, we propose to recognize these environments. If characterizing these situations is easier than characterizing the potential forms of the intruders our approach makes sense.

From the point of view of a perceptual system, the objective is not only to pick up objects moving but also abnormal situations.

Automatizing video monitoring seeks to reduce the need of human intervention, but could it also increase the reliability of human operators ? To answer this question we need to understand the limitations of human operators. Human eyes limit the number of screens one can watch at the same time. Human attention capacity limits the number of monitoring tasks that can be effectively undertaken at the same time (Simons & Chabris [1999]). The results of human operators depend on the complexity of the environment watched, the number of distractors and the frecuency of the events (Rankin *et al.* [2012]). Complex environments (dynamic background, noisy cameras etc.) or a high number of distractors (e.g.: irrelevant events) increase the chance that an operator misses an intruder. The lack of events during a long period of time may impact in the operators attention, and hence in its capacity to detect intruders.

The reliability of human operators can therefore be increased when we reduce the time spent watching video where nothing happens and also when we reduce the number of distractors. An automatic system should show only cameras with high probability to be streaming an intrusion.

4.3.1 Method

4.3.1.1 Implementing an intruder detection system

Our choice to build an intruder detection system is a combination of background subtraction to segment the image and point tracking to model the behavior of the intruder. We have seen that appearance may not be a secure characteristic of an intruder, but the way it moves may be better. Background subtraction by itself does not give information relating objects from one image with the following ones. One way to characterize the way an object moves is to track its position over time. Without tracking, the segments proposed as intruders would always be appearing in the scene. Tracking results can be used to create constraints characterizing intruders (García-Martín *et al.* [2011]), such that a static segment or the ones "moving" not moving in a particular direction are not considered as intruders.

Background subtraction has been chosen over other motion detection techniques because it offers high reliability and performance. Temporal filtering for example is based on temporal differencing (Lipton *et al.* [1998]). This technique uses a thresholded difference of pixel between consecutive images (two or three) to extract the moving object, so it shows high computing performance. However its detection accuracy may be weak, failing in extracting all the relevants pixels of a target object or leaving holes inside moving objects (Kim & Street [2004]). Optical flow is another approximation to image motion defined as the projection of velocities of 3D surfaces points onto the imaging plane of a visual sensor (Beauchemin & Barron [1995]). Different optical flow techniques are detailed in Barron *et al.* [1994], most of them are computationally complex. Another important withdraw is that optical flow algorithms are very sensitive to noise, which is very common in video from CCTV cameras (Hu *et al.* [2004]).

Contour tracking (Peterfreund [1999]) is likely the more reliable approach but with a higher computational cost. Another decision factor is the video resolution. Low resolution makes contour detection complicated. Many sites still have low resolution cameras, which makes point (Salari & Sethi [1990]) or kernel (Hager *et al.* [2004]) tracking more attractive. We choose point tracking using either kalman (Zhong & Sclaroff [2003]) or particle filters (Yan *et al.* [2010]).

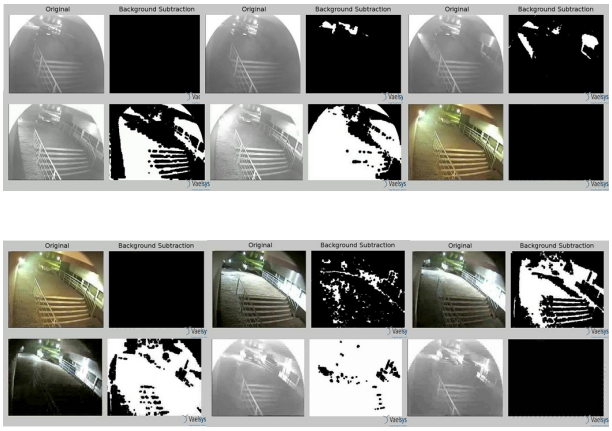


Figure 4.5: Evolution of scene and its BS with illumination from a car

4.3.1.2 Detection of problematic environments

Overview. The ideal intrusion monitoring system would report every single intrusion and only intrusions. When the system misses an intrusion we call it false negative (FN). When the system reports an intrusion, whereas there is none, we call it false positive (FP). The ideal intrusion monitoring system would have zero false negatives and false positives. Table 4.8 represents these definitions.

	Intrusion	No intrusion
Detection	True Positive (TP)	False Positive (FP)
No detection	False Negative (FN)	True Negative (TN)

Table 4.8: Outputs for intruder detection classifier

Brutzer *et al.* [2011] shows that 70% is the best precision for 90% recall using BS. In the worst case, "sudden illumination change" or "light switch" the precision goes down to 10%, even with a recall smaller than 50%. Examples of images with light switching are shown in figure 4.5. None of the presented BS algorithms is able to deal with these complex environments. Actually, even light changes from clouds could be challenging as we can see in figure 4.6.

Analysing motion patterns by setting tracking rules could help, but sudden

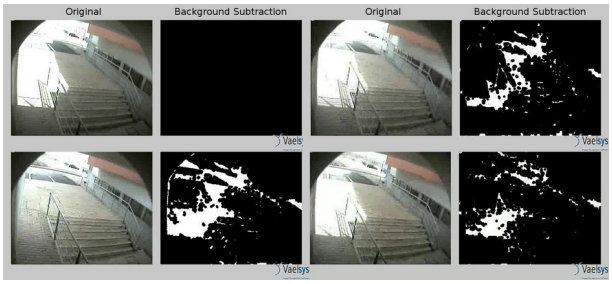


Figure 4.6: Evolution of scene and its BS with illumination from sun with clouds

foreground object appearance and disappearance conveils a big challenge for the dynamic model estimators. If we look at the images from figures 4.7 we observe foreground objects of different sizes and shapes. How would a human explain why illumnation change is not an intrusion when lots of foreground objects show up? Do we need to perceive the objects in a scenery to identify its semantic category? This question is addressed by Navon [1977] when the author suggests that human perception proceeds from global analysis, and this analysis is done before extracting local features, so no object analysis would be required by humans to recognize, for example, the "light switching" event.

These empirical and theoretical statements infer that object based approaches (local) will not be able to handle some common happenings in video surveillance. When replicating human procedure, global feature analysis could give complimentary information to solve some of the local approach limitations. We propose a new method to deal with such environments. It is not intended to improve detection as such, but to improve the end-to-end solution by reducing the number of FP an operator has to check. In a first stage objects are detected using an implementation of an IDS as described above. The results of the first stage are processed and classified. The idea is similar to the one used by people detection methods, first object detection and then a classifier verifies if it is a person, but instead of recognizing people we recognize scenes. We are not interested in recognizing "intrusion scenes" but in recognizing scenes that cause problems: FP.

In order to recognize these "problematic scenes" we use a classifier. In this case a detection is to label a scene as "problematic". But we must pay attention to the difference between the "problematic scene detector" (PSD) and the "intruder

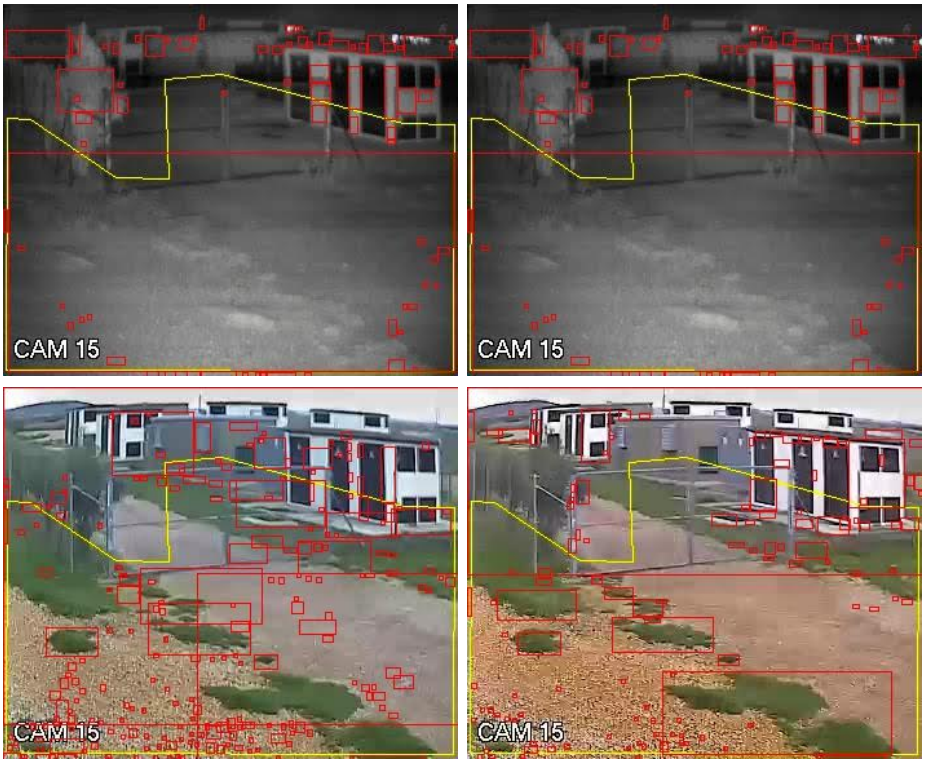


Figure 4.7: Examples of foreground objects generated by illumination change

detection system" (IDS) metrics. In the latter a FP is a wrong intruder detection, while in the former it means a wrong problematic scene detection.

We propose that for PSD, false negatives (not detecting a problematic scene) are less harmful than false positives (classifying as problematic a scene that is not). The reason for this is that "positives" may get lower priority in a review by human operators. In case of doubt we'd rather have the video sequence verified by an operator with regular priority.

Figure 4.8 shows a scheme of the complete system design. Only positives of the first stage of intrusion detection are analyzed. In the second stage positives are labeled, such that an operator can prioritize or even ignore some of the positives reported by the first stage. In the following sections we show how global features could help and a method to use them. This design does not pretend to describe or compare best of class global features nor classifiers. We wish to present a new approach to face false alarms for intrusion detection systems based on video

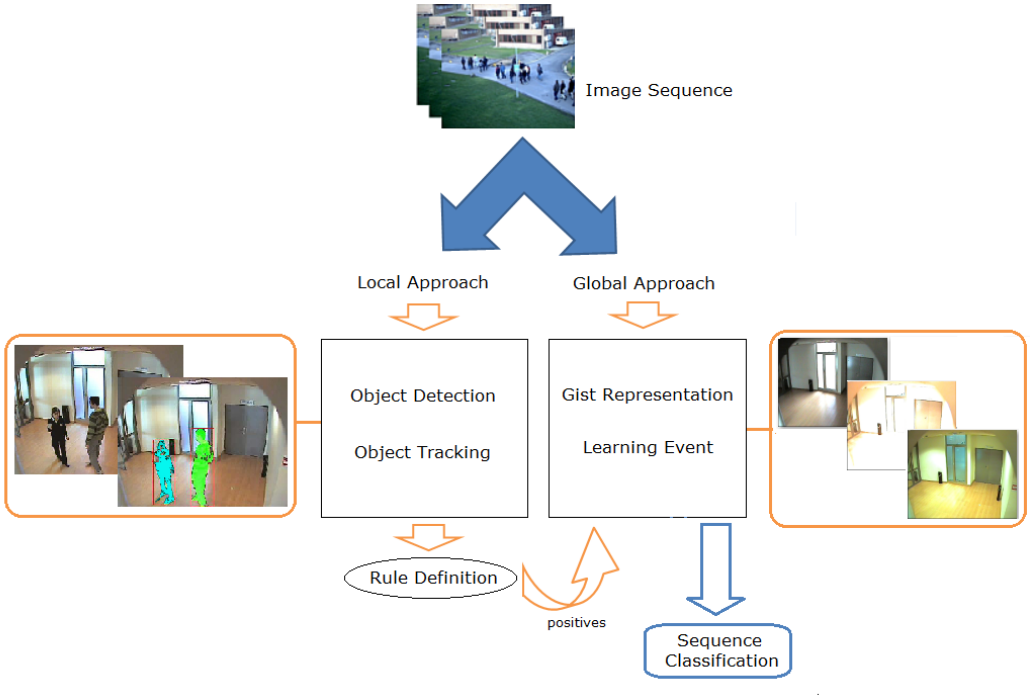


Figure 4.8: System design

surveillance.

Algorithm selection. Our proposal is to use global features to define "light switching" or "sudden illumination changes" scenes and filter them, so that the overall false positive decreases. Different algorithms can be used to extract global features. Simple features like color and texture are recognized to be important features for image representation (Makadia *et al.* [2010]). Color can be represented using "block truncation coding" (Mohammed & Abou-Chadi [2011]), a technique that quantize color information and preserves statistical moments like mean and standard deviation. Haralick *et al.* [1973] describes easily computable textural. The extracted features representing a scene are called "feature vector" or FV.

When an image representation is available for an observed frame or sequence, event recognition, like human actions, become a classification problem (Poppe [2010]). The survey on vision-based human action recognition describes two kinds of direct classification, nearest neighbors for example, and discriminative, for example Support Vector machines (Danafar & Gheissari [2007]). Another example of dicriminative classifier are multilayer perceptrons (MLP) (Duda *et al.* [2012]). In the case of "light switching" or "sudden illumination changes" we want to label video frames or sequences with yes or no "light switching" or "sudden illumination change". Our proposal is to train a classifier like KNN, SVM or a MLP to learn this kind of scenes that should be filtered. Being able to recognize these scenes will help to reduce the number of false positives of the intruder detection system. The procedure would be:

- First an operator detect a concrete false positive that happens often
- Videos of this concrete false positive are stored to be used as training samples
- A classifier is trained to learn this false positive

Based on the previous research we propose to use global features to represent images in order to identify common false positive scenes that should be filtered. Albusac *et al.* [2009] found two main approaches for behavior analysis: (1) deal

with normal patterns in order to detect deviations or anomalous behaviors or (2) define abnormal patterns in order to carry out matching to detect abnormal events. In the examples of "sudden illumination changes" or "light switching" we propose to label these scenes as abnormal and match them with previously learnt examples.

The abnormal pattern or false positive to be detected is called target. The classifier is trained using FVs from target scenes and FVs from non-target scenes. PSD will filter or prioritize the scenes classified as target.

4.3.2 Experiments

4.3.2.1 Evaluation metrics

Detection results may be compared using different metrics. Powers [2011] analyses three metrics Recall, Precision and F-measure. Barnich & Droogenbroeck [2011] and Elhabian *et al.* [2008] prefer to use another one called Percentage of Correct Classification (PCC). Table 4.9 describes each metric.

Recall	$R = \frac{TP}{TP + FN}$
Precision	$P = \frac{TP}{TP + FP}$
F-measure	$F1 = \frac{RP(1+\alpha)}{R + \alpha P}$
PCC	$PCC = \frac{TP + TN}{TP + TN + FP + FN}$

Table 4.9: Evaluation metrics

A common problem for detection systems is that FP and FN are hard to minimize at the same time. The higher sensibility of the detector the higher the probability to make wrong detections (FP), but decreasing sensibility increases the probability of missing elements (FN). In our opinion for an intrusion detection system, FN are usually more important than FP. The probability to detect the intruder must be as close to 1 as possible, even if this has a high FP rate as a consequence. Since PCC assigns FN and FP the same importance we do not recommend this metric to compare intrusion detection systems. F-measure is

probably the best metric because it combines recall and precision in a way that can be biased with a parameter α. α can be used to give more or less importance to the FP rate.

However PCC can be a good metric to compare the performance of a determined intrusion detection system in different sites. Recall and precision depend on TP, and hence if the number of TP of a site is small compared to another site, even if the number of FP and FN remain the same, recall and precision will be lower. On the other hand for PCC, when FP and FN remain the same, the decrease in TP is compensated by an increase of TN, in such a way that PCC remains the same.

Background subtraction comparison are usually evaluated using pixel-based methods (Barnich & Droogenbroeck [2011]; Brutzer *et al.* [2011]; Xu *et al.* [2016]), thus considering foreground detection as a binary classification of each pixel. In this case a FP is a pixel labeled as foreground when it is not, and a FN is a pixel labeled as background when it is not. Therefore we cannot extrapolate pixel-based evaluations of background subtraction methods to intrusion detection systems. As an example, using Brutzer *et al.* [2011] results, when a method presents a recall over 0.9, more than 30% of the pixels labeled as foreground belong to the background. If we consider foreground detection as an intrusion, it is likely that every second several frames have background pixels labeled as foreground, and thus reported as intrusion.

Instead of pixel-based methods we suggest that frame-based or sequence-based methods give a better idea of the performance of an intruder detection system. Frame-based methods consider intrusion detection as a binary classification of frames, labeling each frame as intrusion or no intrusion. Sequence-based methods consider intrusion detection as a binary classification of sequences of video, labeling the global sequence as intrusion or no intrusion. These evaluation methods are used by "people detection" works (García-Martín *et al.* [2011]). Gorodnichy [2005] presents a survey of evaluation datasets used by the scientific community in video analytics in the video surveillance context. The best suited for intruder detection is probably the dataset "sterile zone" from the ILids collection. It is composed of 24 hours of video of people walking, running, crawling or rolling in a grassed area. For our tests we wish to have a dataset more extensive with video

from different cameras and a period comprising many days.

4.3.2.2 Intruder detection using state-of-the-art methods

We have chosen 3 sites with a total of 76 analog cameras, table 4.10 shows the number of cameras per site. Each site has a 5-6 year old DVR recording at 353x288 of resolution, six frames per second (FPS).

Site 1	Site 2	Site 3
25	26	25

Table 4.10: Cameras per site

We implement an IDS using Running Gaussian Average for segmentation and a Particle Filter for tracking. We set three simple rules for intruder detection: it must be bigger than four pixels, has to be tracked at least in three frames and be within a wide area defined by a yellow polygon. We process 19 days of video, thus 34,656 hours.

A "positive" is a sequence of video reported as intrusion. When two frames are classified as intrusion, a 10 seconds video clip is recorded, so the maximum number of positives per minute is 6, thus 360 per hour and more than 12.476.160 for all the cameras during the 19 days of experiment.

4.3.2.3 Detection of sudden illumination changes

In order to test our proposal to reduce false positives using a global approach we picked twenty videos per day of camera 15. This camera is a clear example of "sudden illumination change" that repeats over a period of time. Table 4.13 shows that this camera generates more than 90% of the positives of all the cameras during the period of three days. The first day the number of positives from camera 15 was two order of magnitude bigger than the number of positives generated by the second camera with more positives. If we look at the timing, we noticed that the IDS is generating positives continuously for hours. After watching several video clips we notice that this camera is somehow damaged, the

Figure 4.9: Frames from false positives videos in camera 15

night mode generates an effect of random "sudden illumination changes". Describing illumination changes with words is not simple, figure 4.9 shows two series of frames from two of the problematic days illustrating the video streamed by the camera. The problem stops after the third day.

Ten contiguous videos (positives) were picked between 21:00 and 22:00 each day to form group A of videos, referred as D1A, D2A, D3A to identify day and timing. Then ten contiguous videos were picked between 23:00 and 24:00 hours each day for group B, referred as D1B, D2B, D3B. So each group has thirty video clips of scenes where the IDS had reported intrusion, while none of them had any intrusion event. We have a third group (DN) composed of thirty videos extracted from recordings of the same camera in a day without sudden illumination changes. These videos are almost indistinguishable from a visual point of view. None of the videos of DN were labeled as intrusion by the IDS, since they represent no intrusion. We use DN videos to validate our proposal.

The idea is to learn sequences with events that should not have been treated as positives by the IDS and filter them with a PSD. In order to learn these scenes we encode each scene's color and texture features as described in section 4.3.1 into a feature vector FV.

Videos are subsampled to 1 FPS, we eliminate the two first frames that were reported not to have intrusion by the IDS and discard the ones in excess of 11. Then a binary multilayer perceptron (MLP) is trained to classify the FV, into "target" or "no target".

In the first group of experiments we have three sets of target videos: D1A, D2A

and D3A. We train MLPs for each set of target videos with sets of videos from group B (different hour) and group DN (no illumination change). For example D1A is tested separately with D1B, D2B, D3B and DN. The number of frames in all the sets is 110 (10 videos x 11 frames), except DN, which has 330 (30 videos x 11 frames). The training group is made with 75 % of the frames from the target videos, while the testing group is made with the videos of the non-target set.

In the second group of experiments consider each day to have a different class of illumination change. We wish to recognize sequences of D1A and try if we can differentiate them from sequences from D2A and D3A. Then we change the target to D2A and finally to D3A. When target is D1A, 75 % of the training set come from D1A, while 25 % come from D2A and D3A. The same distribution is found when target class is D2A and D3A. We guarantee that each class has its 75% in the training group and 25% in the testing group.

In the last experiment we consider all the illumination changes as one class. Training is done with half of the sequences from group A (D1A, D2A, D3A) as target and half of the sequences from group DN as non-target. Testing is done with the other halves.

4.3.3 Results

4.3.3.1 Intruder detection using state-of-the-art methods

Table 4.11 describes the positives and PCC per site. PCC is calculated using the formula described in section 4.3.2.1. We consider a true positive a sequence where a living body is moving in the defined area. Figure 4.10 shows examples of true positives sequences with rabbits, a bird or someone in a car. FP are positives due to illumination changes.

Counting the number of FN was not affordable, because it would have required reviewing thousands of hours of video but after analyzing the kind of intruders that were detected (tiny rabbits or birds), much smaller than humans, we could assume without much risk that we had zero false negatives. Even if this assumption is somehow wrong it wont affect the following experiments where we test our proposal since we aim to filter false positives. The rest of potential sequences are counted as TN.

Figure 4.10: Examples of frames with real elements detected

	Site 1	Site 2	Site 3
Total	1186	1404	6027
Animals	471	765	879
People	523	300	317
Illumination changes	192	339	4831
PCC	99.995 %	99.992 %	99.882 %

Table 4.11: Distribution of positives and PCC per site

Site 3 has an order of magnitude more positives generated by illumination changes than the other sites. Looking into the detail, we observe that most of them are generated during 3 consecutive days (D1-D2-D3). The following results are extracted from these days.

Table 4.12 shows the number of cameras from site 3 with zero postives, one to four, five to nine and ten or more positives. Each day, more than 76% of the cameras reported less than five positives.

Table 4.13 shows the distribution of all the positives reported. The first day more than 97% of them were generated by a single camera (number 15). Day two 95% and day three more than 65%. If we take out the three cameras with more positives (12% of the 25), we would reduce positives at least in 85% and up to 98.74%.

	0	1-4	5-9	>=10
Day 1	24%	44%	12%	20%
Day 2	60%	16%	12%	12%
Day 3	52%	24%	4%	20%

Table 4.12: Proportion of cameras with positives (true and false) in the intervals from site 3

	Total positives	1 cam.	2 cams.	3 cams.
Day 1	3738	97.96%	98.34%	98.74%
Day 2	861	95%	97.32%	89.56%
Day 3	244	65.16%	79.91%	85.65%

Table 4.13: Positives reduction subtracting most problematic cameras (1 cam. corresponds to camera 15)

4.3.3.2 Detection of sudden illumination changes

Table 4.14 shows the results of the first group of experiments. Three different target classes are defined and each one is tested with four sets of videos considered to be non-target. When a non-target video is correctly classified we report it as a rejection that could be filtered or prioritized before showing it to an operator. High rejection means that the PSD considers the non-target videos to be different from target ones. We observe high rejection in all the cases except on videos from the same day.

In the second group of experiments we consider different days to have different kinds of illumination changes. Table 4.15 shows the results of the PSD trying to detect group A videos from D1, D2 and D3. More than 90% of the frames are correctly classified.

The results of the classifier trained with group A as target shows a 100% of rejection of non-target videos from DN. If all the FP caused by sudden illumina-

Target / Non-target	D1B	D2B	D3B	DN
D1A	**11.01%**	94.52%	100%	100%
D2A	97.45%	**23.74%**	95.38%	100%
D3A	99.15%	77.69%	**6.92%**	100%

Table 4.14: Rejection rates of non-target videos

	D1A	D2A	D3A	Mean
Test Error	1.22%	7.31%	8.53%	5.68%

Table 4.15: Sequences misclassified when training and testing with videos from group A (same hour different days)

tion changes are removed, the amount of FP in all the cameras would be reduced to 729. Table 4.16 compares the precision of IDS with and without PSD. Results are calculated using data from all the cameras, with an α value of 1.

	Recall	Precision	F1
IDS	1	0.38	0.55
IDS+PSD	1	0.81	0.90

Table 4.16: Metrics of intrusion detection solutions

4.3.4 Discussion

4.3.4.1 Intruder detection system using state-of-the-art methods

We have implemented an IDS with state-of-the-art techniques. Manual sampling suggests that it is able to detect the presence of any animal or person, and therefore has an insignificant number of FN. If we presume FN to be zero, then IDS

reaches a recall of 1 in all the sites, which is the ideal case. However precision is lower than the one from other methods, such that the F-Measure, which combines both, recall and precision, is below state-of-the-art results. Our IDS is not as robust as other methods in the literature but is faster (Piccardi [2004]). If we look at PCC, all the sites present a measure over 99.88%. This result is important because it shows that for perimeter protection even a simple method designed to guarantee a high recall, reports less than 0.07% of the video as positives.

In the context of perimeter protection, the problem of evaluating intruder detection system as if it was a people detection system is that we are forgetting that more than 99.9% of the time there is no people in the scene. If among one million sequences of 10 seconds of video, there were only two with people, and the system would have wrongly classified one of them and nine other sequences, the F-Measure would be below 0.2. Whereas if we consider intrusion detection as a problem of background modeling, in the same example the F-Measure would be above 0.99.

Different reviews of background modeling show how different techniques are able to adapt to several changes. However, sometimes these changes are just too extreme to allow the model to adapt. Sudden light changes like the ones seen in camera 15 from site 3 are an example. Day 1 the PCC goes down to 57.61%. The same camera, Day 3 has a PCC of 98.17%. To understand what these numbers mean we can imagine that the difference between 100% and the PCC is the time an operator spends with a camera. If we suppose that an operator can only verify one camera at the time, in the case of PCC = 57% almost half of his time (43%) would be spent on just one camera. It does not seem sustainable. Even a PCC of 98% is hard to sustain for any monitoring center, since 50 cameras would take all the operator's time. To be profitable, monitoring centers require operators to manage hundreds of cameras. In order to avoid monitoring center collapse, common practice is to inhibit problematic cameras. this practice is supported by results: inhibiting the three cameras with more positives would make the PCC jump to 99.98%.

4.3.4.2 Detection of sudden illumination changes

Instead of leaving aside problematic cameras, an intelligent system should learn to identify the problem, such that the video coming from those cameras could be further processed or at least be classified and prioritized. In this work we have presented an alternative to camera inhibition: detecting problematic scenes.

In order to test its accuracy we executed different validation tests with FV from sequences that have no frame in the training group. Results from table 4.14 show that a trained machine to learn D1A video sequences, rejects not only 100% of frames from videos without sudden illumination changes (DN) but also 94.52% and 100% of frames from D2B and D3B video sequences. Something similar happens when training to learn sequences of class D3A: 99.15% and 77.69% of the frames from other classes are rejected. Training to learn D2A video sequences also shows very high rejection percentage with frames of D1B 97.45% and D3B 95.38%. A high rejection rate means that the system will not filter alarms it has not learned, and thus will limit the risk of increasing the false negative rate of the overall system.

If we look at sequences from the same day, table 4.14 shows that the rejection rates are much lower, from 6.92% and up to 23.74%. Watching video from D1, D2 and D3 shows that videos from different days are in fact different from a visual point of view. The nature of the event is the same, a damaged camera, but the consequences have visual differences, illumination changes does not look exactly the same. The proposal seems to be specific enough to differentiate between different kinds of illumination changes. This hypothesis is reinforced by the results from table 4.15. We are able to differentiate illumination changes from one day and another with an accuracy up to 98.78 %.

The generalization test has proved to have been successful. The method is able to learn the concept "sudden illumination change" from a combination of the different illumination changes from D1, D2 and D3. This is a strong point of our approach since it can be used as a specific classifier or as a generic one. More errors appear when video sequences have higher visual likeness, so the more visual differences there are between illumination changes, the better the classifier works.

PSD can be applied to classify, filter or prioritize detections of an IDS. PSD should detect scenes belonging to previously trained classes. Errors in this classification have two possible consequences. The PSD FN, scenes that should be filtered but which are not, are counted as a FP for the overall solution. The PSD FP, scenes that should not be filtered, intrusions, but are wrongly classified create new FN for the overall solution. The results from our experiments suggest that our implementation of a PSD method can be very specific, detecting only sequences very similar to the ones learned and therefore keeping the number of FN of the overall system close to the one of the IDS.

4.3.5 Conclusions

State-of-the-art methods allow to build real-time intruder detection systems with extremely high accuracy, so that one single operator could verify hundreds of cameras. The level of attention of a computer is not comparable with the one of a human. However empirical tests show how a single camera could saturate a monitoring center with thousands of alerts caused by faulty light adaptation system. Even when this kind of events would be rare, and statistics of state-of-the-art methods would make us feel comfortable, when such an event occurs the processing of this camera has to be turned off, either to ignore it or to watch the video continuously.

Instead of trying to increase robustness for environments where modeling is hardly possible, we propose a method to solve these problematic happenings based on global features to learn scenes. We start from the observation that, false positives (detections that should not be reported) are rare but when they happen they do so with intensity, repeating in short periods of time (minutes to hours). Being able to identify the first false positives, something a human operator could easily undertake, then we could train an automatic system to recognize the following ones and filter or prioritize them.

Global approach seems to fit better with events that are hard to be defined using local features, which makes a lot of sense (Navon [1977]). However the definitions (scene classification) we presented in this paper are only based in visual likeness, while a human would rather prefer semantic definitions. “Sudden

illumination changes" for example has a clear meaning for humans but may have different representations in video. In our experimental events of class D1, D2 and D3 could be labeled with "sudden illumination change" even if their visual representations are different. Some categories are hardly defined with high level features. Categorizing the parts is sometimes a waste of resources.

The present method can be very specific so that it differentiates between different types of illumination changes or it can be more generic to represent a broader definition of a sudden illumination change. Specificity in the PSD is good for the overall solution since it minimizes the number of FN. We assumed that the IDS has a FN rate close to zero. A PSD with high precision will filter IDS FP without adding FN to the overall solution, thus keeping its recall high. On the other hand generalization in the PSD may be useful to reduce the number of scene types, categories, that need to be learned. Our results are good in both, specific and generic experiments, so we do not need to decide which one is more important, however other kind of problematic scenes could require to do so.

The probability of intrusion is not increased by the fact that illumination changes. A system with limited resources should give priority to the cameras with higher chances of being capturing a suspicious scene, the information from the whole scene could be used to avoid giving priority to cameras with illumination changes. Like Navon [1977] and Torralba *et al.* [2006] we believe that the analysis of global features may offer relevant information to guide the process of perception.

4.4 Aesthetic perception

The fourth example is about calligaphy perception and is based on Perez *et al.* [2014]. The word calligraphy comes from the greek "kallos" (beauty) and "graphe" (writing): "the beauty of writing". Addressing art from a computational point of view could be surprising since computers are the paradigm of logic and rationality, while art is related to emotions. However in the last few years there is an increasing interest in a topic called "affective computing" which proposes to give computer the ability to have emotions.

Picard [2000] presents models for human emotion recognition, computer-assisted learning, perceptual information retrieval as well as arts and entertainment. Tao & Tan [2005] defines affective computing: "trying to assign computers the human-like capabilities of observation, interpretation and generation of affect features". If we look in the literature for writing art generation and interpretation we find several works about Chinese and Arabic calligraphy. Zhang *et al.* [2013] presents a method for calligraphy style recognition based on global features and support vector machine training. Moustapha & Krishnamurti [2001] generates arabic calligraphic compositions by manipulating symmetric changes and analyzing its visual effects. Xu *et al.* [2005] presents an intelligent system which is able to generate a great variety of stylistic calligraphic characters. In order to select the more pleasing characters Xu *et al.* [2007] introduces a neural-network algorithm that is able to select the more pleasing ones. However the method requires the intervention of humans to decompose a character into strokes, we find once again the problem of segmentation. Segmenting characters in strokes is important because Xu *et al.* [2007] grades calligraphy by grading individual strokes and then spatial layout of strokes.

We instead want to avoid complex segmentation, and thus will work directly with the handwriting. Instead of dealing with high-level features, like characters, we will use a set of low level features and test if a perceptual system is able to find a set of features that represent people's opinion about calligraphy.

4.4.1 Method

The objective of the pre-processing phase is just to clean the image. Cropped images are binarized and complemented, such that handwriting becomes white and non relevant information black.

The result of the preprocessing phase is a matrix of white and black pixels (figure 4.11) that will be used as input for different algorithms in order to extract shape feature that will help to describe the writing in a way that a classifier can simulate the taste of a human. Pham [2000] presents a general framework for constructing shape aesthetic measures, which "has been achieved by drawing knowledge on how to produce aesthetic products from a number of fields to obtain

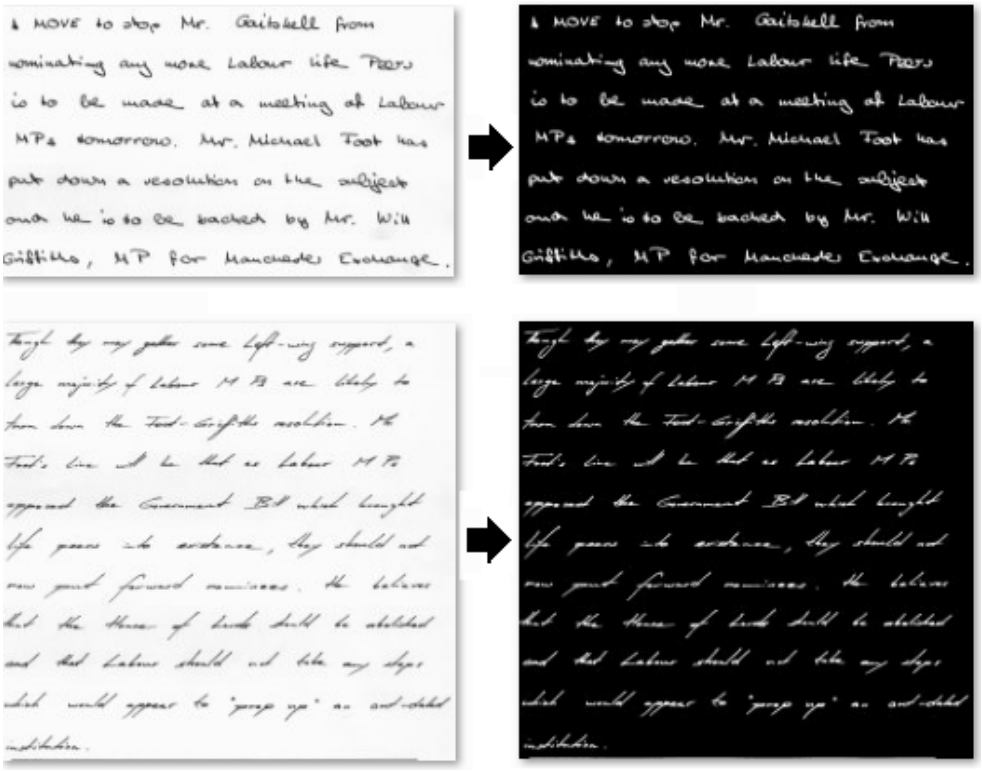


Figure 4.11: Example of complemented images

a rational view of aesthetics". It groups basic principles for designing aesthetic painting, graphics, etc. in four categories: opposing forces, resolution of conflicts, movement and global impression. Pham [1999] describes which shape features (called variables) influence in these aesthetic principles:

- Opposing forces: balance, contrast, proportion
- Resolution of conflicts: dominance, harmony, composition
- Movement: rhythm, gradation, dynamic
- Global impression: simplicity and solidity

Yang *et al.* [2008] compiles several shape feature extraction techniques we found useful to compute most of the variables described in Pham [1999]. For this work we use the following measures:

- Mass (m): the number of pixels in the image that contain handwritten text. This measure is normalized to the size of each image and it is in the range [0,1].
- Center of mass (x_c, y_c): is the unique point where the weighted relative position of the distributed mass sums to zero. This measure is normalized to the size of each image and it is in the range [0,1].
- Eccentricity (ec), is the ratio of the distance between the foci and the major axis length of the ellipse that has the same second-moments as the object. The value is between 0 and 1.
- Orientation (θ), is the angle the angle in degrees between the x-axis and the major axis of the ellipse that has the same second-moments as the object. The value is in the range [-90, 90].
- Euler number (ε): scalar that specifies the number of strokes in the text minus the number of holes in the text. A hole is defined as a space inside a stroke.

- Solidity (so), is the ratio of the mass of the text and the area of the convex hull of the text. The convex hull is defined as the minimum convex perimeter that can contain the text.
- Extent (ex), is the ratio of the mass of the text and the area of the bounding box that contains the text.

Figure 4.12: Calculation of the Euler number on different shapes

The bounding box is the smallest rectangle containing the text. We also added the following simplified features

- Font-size (sz), small, big or variable.
- Text inclination (yes / no)

Inspired by Pham [1999] variables we relate opposing forces with mass and center of mass. Resolution conflicts with eccentricity and orientation. Movement with Euler number, inclination and size, and global impression with solidity and extent. These ten features are stored in a feature vector FV which will be used to represent an image. So we will have as many FV as images.

FV = [m, x_c, y_c, θ, ec, ϵ, so,ex, sz, in]

Image classification is done using an instance-based learning algorithm: k-Nearest Neighbors algorithm (K-NN) (Cover & Hart [1967]). This kind of algorithm stores "in memory" patterns from the training set and compares them with testing patterns. The comparison is done using the Euclidean distance. A random group of images is selected as training group. One or more persons label each training image with "beauty" or "ugly". For every test image, K-NN classifier is used with its corresponding FV to decide if it is "beautiful" or "ugly" writing.

4.4.2 Experiments

4.4.2.1 Database description

The experiments are done using images from Marti & Bunke [2002]. This database contains forms of different handwritten English text which were scanned at a resolution of 300dpi and saved as PNG images with 256 gray levels. It was first published in Marti & Bunke [1999] at the ICDAR 1999. For our experiments we picked 1051 samples from the database. The same text is written by different authors, each scanned in a different page. The same authors also wrote different texts, and again each is scanned in a different page. Figure 4.13 shows different scanned pages of different texts from the IAM Database.

Figure 4.13: Samples from IAM database

4.4.2.2 Human calligraphy evaluation

The classification between "beautiful" and "ugly" for the writing is done by questioning two people, and based on their sole opinion. No further instruction was given. Figure 4.14 shows two examples of this classification. Text content is in English, a language not spoken by the questioned people. This helps to avoid influences from the content on the evaluation of the visual aesthetics.

A sample of writing is qualified as "beautiful" only if the two people classified it like that. The same process is used to classify "ugly" images. Controversial images, the ones that were classified as "beautiful" and "ugly" are removed. 283 out of 1051 (27%) were removed using this criterion. The rest of the dataset is divided in two groups. The first group (training) is a random selection of 384 images. The second group (testing) is composed of the 384 images remaining. Table 4.17 shows the composition of each group.

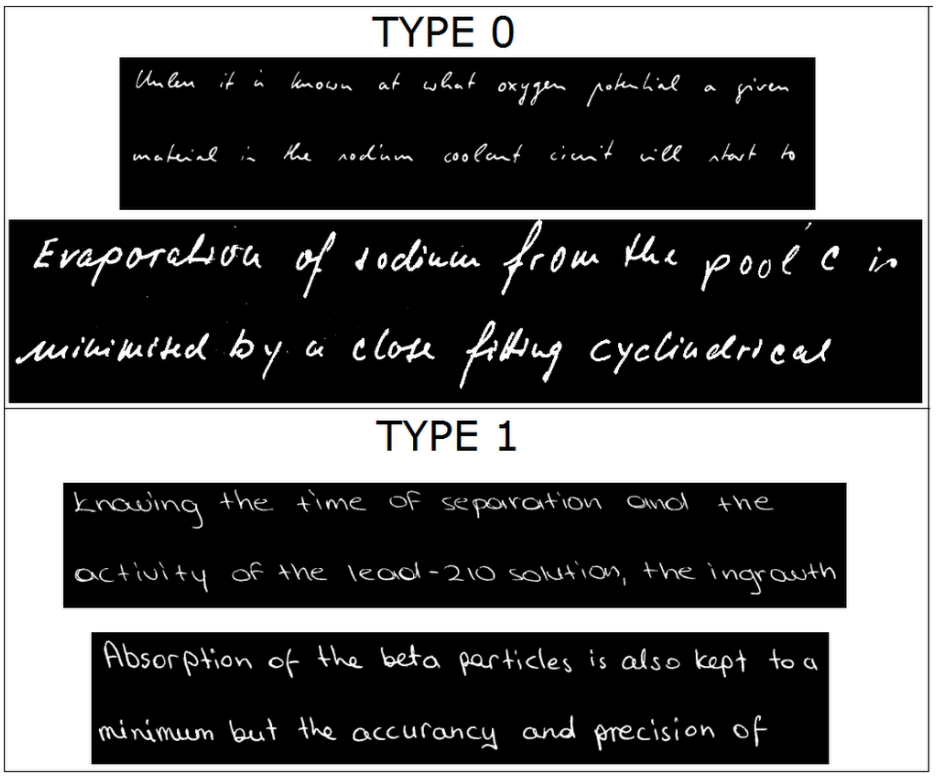


Figure 4.14: Type 0 = Ugly ; Type 1 = Beautiful

Name	Type 1(Beatiful)	Type 0 (Ugly)
Training	114	270
Test	114	270

Table 4.17: Composition of "ugly" and "beautiful" groups

Name	m	x_c	y_c	ϵ	θ	E	so	ex	sz	in
FV	X	X	X	X	X	X	X	X	X	X
FV_1						X	X	X	X	X
FV_2						X	X	X		X
FV_3						X	X			X
FV_4	X	X	X	X	X					
FV_5	X	X	X			X			X	X
FV_6	X	X	X				X	X		
FV_A						X			X	X
FV_B	X					X		X	X	X

Table 4.18: Feature vector composition: m = mass; $x_c y_c$ = center of mass coordinates; ϵ = excentricity; θ = text orientation; (ε) = Euler number ; so = solidity; ex = extent; sz = font sixe; in = inclination

For each image we create its FV as described in section 4.4.1. In order to better understand the importance of each feature we create subvectors with a selection of features from the main FV. Table 4.18 describes the selected features for each subvector type: $Fv_1 ... Fv_6$.

Fv_1, Fv_2 and Fv_3 were generated using a ranking algorithm to select features. We used the same model than Guyon *et al.* [2002] for gene selection. A SVM is used as estimator to assign weights to features. The goal is to select features by recursively considering smaller and smaller sets of features. First, the estimator is trained on the initial set of features and weights are assigned to each one of them. Then, features whose absolute weights are the smallest are pruned from the current set features. That procedure is recursively repeated on the pruned set until the desired number of features to select is eventually reached.

FV_A and FV_B were generated for evaluator A and B using the same procedure to select the features with relevant information. Features are considered relevant

when their associated weights are positive.

Fv_4, Fv_5 and Fv_6 were generated using the relation stated in section 4.4.1 between shape features and aesthetic principles. Fv_4 represents the features of "opposing force" and "resolution conflicts". Fv_5 represents the features of "opposing force" and "movement", while Fv_6 represents the features of "opposition force" and "global impression".

4.4.2.3 Automatic calligraphy evaluation

The classification of each feature vector is done using the K-NN algorithm. Testing FVs are classified using training FVs. Testing Fv_1 is classified using training Fv_1 and so happens with $Fv_2...Fv_6$. If the class proposed by the algorithm for a feature vector is the same that the one the two people fixed for the corresponding image we consider a success. If not it is considered a failure. We tried 1-NN, 2-NN, 3-NN, 4-NN and 5-NN. The best performant turned out to be 3-NN.

4.4.3 Results

Table 4.19 presents the results obtained in the tests using different feature selections, coded in different feature vectors (FV, $Fv_1...Fv_6$). In each case we show the global success rate (SR) calculated by dividing success classifications by the total of samples. Then the "beautiful" classification success rate (B) calculated by dividing success classifications of "beautiful" writings by the total "beautiful" images. The same calculation is done to get the "ugly" classification success rate (U) with "ugly" writings.

4.4.4 Discussion

Table 4.17 shows that 228 were classified by people as "beautiful" while 540 were classified as "ugly". The proportion is 29,7% of "beautiful" versus 70,3% "ugly", so a dummy classifier, labeling all the images as "beautiful" will be successful in 29,7% of the cases. If all the images were labeled as "ugly" the successful rate would be of 70,3%.

	Global Success	B (Type 1)	U (Type 0)
FV	74.47%	50.00%	84.81%
FV_1	**79.42%**	**60.52%**	**87.40%**
FV_2	75.52%	52.63%	85.18%
FV_3	77.08%	54.38%	86.66%
FV_4	63.80%	31.57%	77.40%
FV_5	76.82%	53.50%	86.66%
FV_6	64.58%	26.31%	80.74%

Table 4.19: Results for different Features Vectors compositions

"Ugly" writing classification minimum success rate is 77,4% and reaches a maximum of 87,4%. So that every automatic classifier is better than the best dummy classifier (70,3%). Looking at the "beautiful" writing samples we see that 6 from 7 classifiers have better results than the dummy classifier (29,7%). In the best case, the automatic classifier is able to double the success rate with 60,52%. The best classifier for "beautiful" images is also the best classifier for "ugly" ones. This result suggests that the feature selection for this classifier is the one that better represents aesthetic principles.

We have presented two approaches for feature selection. One that tries to represent knowledge (*wissen*) about aesthetics and another one automatic, based on the information learned from samples. The second one, which can be assimilated to knowledge by acquaintance (*kennen*) clearly shows better results. The representation of knowledge (*wissen*) might be questioned, maybe the aesthetics principles are not properly represented by the combination of shape features of FV_4, FV_5 and FV_6. What is clear is that finding a representation based on knowledge by acquaintance is easier. In fact the different feature selection for each evaluator suggests that each person's opinion should be represented by a different feature selection.

Table 4.19 shows that 5 of the 7 tests performed better than 74%. In order to have an idea of the importance of this error we can compare it with the proportion of images that got different labels from the two people in the experiment: 27%.

4.4.5 Conclusions

The "opinion" of 5 classification systems differs less from the human opinion than the opinion between the two people collaborating with the tests. Of course we would need further testing with more people and more opinions per person to claim statistical evidence, but these experiments suggest that machines can be trained to perceive subjective things, such as art, with results similar to humans.

In this example, again, low level features without signification have proven to be more characteristic than high level features based on propositional knowledge.

4.5 Summary

Segmentation is a recurrent problem for different visual perception applications. Dividing an image exactly how a human would do it is a challenging task and often computationally expensive. However simple segmentation methods can effectively be used to achieve good results. Describing people evacuating and people dispersing has proved to be hard. Two persons classify the same sequence of frames differently. However we can make a machine learn a definition of evacuation and dispersion so that it is able to classify video sequences accurately and avoid the complex task of segmenting a crowd.

Perceiving the authorship of a handwritten document may seem a more complex task than recognizing people walking or running. Experts build codebooks with distinctive characters to support the task of writer identification. Segmenting handwritten words into letters is an extremely challenging task for a machine. However writer identification can successfully be achieved with the analysis of handwritten connect components, which can easily be segmented. Furthermore, for a small group of documents, the separation between the connected components has proved to be sufficient to identify writers, without connected component analysis. For a larger group of writers, exploring the space of results of a segmentation algorithm applied with multiple parameters improves the ones given by simple segmentation. Perception is more accurate when more information is gathered, even if the algorithm to extract it is the same with different parameters. Analyzing the parts and combining global and local information can also improve

perception.

In some cases the parts of a whole are not recognizable and therefore a global description is required to categorize the event. Sudden illumination changes are an example of such situations, in which describing the parts does not give useful information. A global description based on low level features is good to define both "sudden illumination changes" as one category and to define different sudden illumination changes as several categories. Global descriptions can be specific or generic.

Finally we have analyzed a subjective problem, aesthetic perception. Machine learning techniques have shown much better results than theoretical attempts to define "beauty". The way of perceiving subjective categories like "beauty" is similar to the way of perceiving authorship or actions. If we agree with Purves & Lotto [2003] theory of vision, it makes a lot of sense.

> the visual system is not organized to generate a veridical representation of the physical world, but rather is a statistical reflection of visual history (Purves & Lotto [2003] p.227).

Chapter 5

Conclusions

The present PhD dissertation seeks answers for the main questions about machine visual perception: (1) *can a machine perceive what a human perceives?* and (2) *how can a machine achieve the results of human vision?*. The answer of the first question requires understanding what a human perceives. The answer to the second question should present the requirements that a perceptual system must satisfy in order to achieve the results of human vision. The question is different to a third question, *how do we build a machine able to perceive what a human does?*. The answer to this question would present the implementation of a perceptual system or its modules. However it would not tell why the selection of these modules is appropriate. If we wish to emulate or improve the results of human vision, we need to understand what should be computed and why before worrying about implementation.

5.1 Human and machine vision

5.1.1 Can a machine perceive what a human perceives?

The short answer would be yes it can. Theories for human vision propose that we see shapes (Marr [1982]), affordances (Gibson [1986]), or a probabilistic representation of the past (Purves & Lotto [2003]). All of them can be represented by relations. Shapes can be represented by geometric relations of pixels. Affordances can be represented by semantic relations, relations between a sign, a shape for

example, and a concept. A concept can be represented by the relation of its properties. Perception is about relations, so that a machine will be able to perceive, if it is able to deal with the different kinds of relations involved in visual perception. In chapter 3 we show how each of them can be represented and computed by machines. In chapter 4 we present examples of a variety of computer vision applications. The results of human visual perception can be emulated in fields as different as activity, handwriting authorship, intrusion or aesthetic perception.

When several elements are related they are given a form and what is given a form is etymologically information. Information is therefore the key element for visual perception. Depending on the nature of the elements, we have different kinds of information. We have proposed to group them in two types: intrinsic and extrinsic information. The former are relations between intrinsic elements of the image, pixels, while the latter are relations between extrinsic elements, knowledge.

We have noted that perception results among humans are not homogeneous, two people can perceive the same object differently. The images 2.9 illustrate the statement from Purves & Lotto [2003]: "the output of any detector to the rest of the visual system is necessarily as ambiguous as the stimulus it presumably encodes" (p.5). The results of any visual system, human or artificial, are not certain, but stochastic. When we see something in an image we consider that we have sufficient evidence to affirm that that something is represented in the image. When a machine categorizes an image, it considers that sufficient evidence has been found to make such categorization.

The categorization of an image depends on the perceptual system that categorizes it, either human or artificial. The results of visual systems should not be compared by the strength of the evidence collected, but by the quality of the definitions used to categorize. For example does the painting from Velazquez 3.1 represent the Corona Borealis? We could not tell without evaluating the definition that leads to such categorization. In order to make a machine emulate the results of human vision we need definitions equivalent to the ones used by humans to categorize images. The results of experiments in sections 4.1 and 4.4 show how a machine can emulate the categorization of a particular person with higher accuracy than another person.

5.1.2 How can a machine achieve the results of human vision?

Computers are powerful tools for information processing, they can store massive amounts of information and perform many types of computation much faster than our brains. Therefore computers seem well suited to implement visual perception systems. The challenge is to find the appropriate computations to perform visual perception. This task is challenging because most of the knowledge required in visual perception is knowledge by acquaintance, which is not the kind of knowledge that a human can express with language.

Low level features. Despite the fact that vision is something natural to people, describing objects that are seen might be complex, even for simple objects. How could a person describe a cat, such that it can be recognized in its different poses and not be confounded with a dog? Descriptions given by humans are based on high level features, for example a cat has four legs, two pointy ears, mustache etc. However definitions made of high level features are usually weaker than the definitions based on low level features, since the former are based on the latter. High level features are just a subset of the possible relations between low level features, which have been conceptualized, but this does not imply that other sets of relations between low level features are not more characteristic for the object.

Experiments in chapter 4 show how systems based on low level features are able to emulate or improve the results given by humans. The features that a person would use to describe an object may not be the best choice to create a definition for a computer. Categorizing and describing are different actions. A high level description is not necessary to categorize an image, but is the common procedure by which people transmit knowledge about recognition because it is easier to express with propositions than low level features.

A search problem. The strategy that we have proposed is to approach visual perception as a heuristic search problem, in which information is gathered at every step to guide the process of search. Like any other system, visual systems have limited resources and in many real world situations the amount of information

available would largely overload them. A heuristic strategy can produce good results with less resources than approaches based on exhaustive search. In the worst case the former can be equivalent to the latter, but an efficient visual system should be able to find targets in many situations without exhaustively scanning the image. In human vision, visual attention is the mechanism by which only a fraction of the available information is processed.

Moreover gathering information may also be a form of verification. In section 3.3.5.3 we present results showing that even state-of-the-art classifiers can easily be fooled. Exploring several approaches leads to a variety of evidence that compensates the weaknesses of isolated classification schemes.

Intelligent agent. The search is guided by the information gathered from the image, the percepts, and the information known by the system, prior knowledge. A well known paradigm in AI to study search problems are intelligent agents. "For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has" (Russell & Norvig [2014] p.38). In this paradigm perception is usually associated to sensors. However we suggest that only by considering perception as the goal of the agent could we emulate the results of human vision. Given a particular image, a sensor will always generate the same result. A perceptual system built as an intelligent agent can perceive the same object as a weapon or as food, for example an apple, depending on the percept sequence and its knowledge. If the system has recognized the action of "throwing", the apple would rather be considered as a weapon and not as food.

With this approach, the result of visual perception is not a label, a term without signification, but the evolution of the perceptual system. Perception changes the percept list, but also the knowledge of the system. What has been seen, can be extracted by evaluating the status of the system. An external system could make queries to the visual system by evaluating if its status satisfies the constraints that define a category. The same status can be queried several times, with different category constraints, so that external systems could get information about the image. The status of the visual system would encode the equivalent

to the Visual Short Term Memory (VSTM). The categories activated within the visual system at every moment could also be related with external systems and be seen as instant perception. Instant perception might trigger a reaction outside the visual system or be ignored, it fulfills the traditional role assigned to sensors.

Knowledge. Unlike sensors, perceptual systems learn and adapt, they acquire knowledge. Knowledge representation is one of the main keys that have revolutionized the field of computer vision in a few years. Deng *et al.* [2009] introduced a dataset with millions of labeled images, Imagenet, which represented a never seen before warehouse of knowledge by acquaintance. This knowledge was represented by a model inspired in biology: convolutional neural network (CNN). CNN had already been used for image recognition more than twenty years before (LeCun *et al.* [1990, 1995]), but with an architecture of around ten thousand connections. To represent the knowledge of Imagenet Krizhevsky *et al.* [2012] uses an architecture with sixty million parameters, whereas Simonyan & Zisserman [2015] goes beyond a hundred million parameters.

5.2 An active system

5.2.1 Top-down and bottom-up processing

Our approach to visual perception combines top-down and bottom-up processes. The key operation for this is segmentation, when and how to divide an image. The principle is to segment as few times as possible and use first segmentation methods based on simple constraints leading to a bounded number of segments. In sections 4.1, 4.2, 4.3 we have seen examples where simple segmentation schemes achieve good results, even when the results of segmentation are not those that a human would select.

Before segmenting an image or a region the system should extract as much information as possible. Direct categorization is performed by computing global and local features. Global features represent statistics about all the elements of the image while local features represent relations between neighboring elements

without need of segmentation. The information gathered from direct categorization is useful to decide if it is worth it or not the division of the image or region into regions or subregions. This process is the top-down component of the algorithm proposed for visual perception.

On the other hand the results of categorizing the parts can be used to categorize the whole, in a process called comprehension. This is likely the first option that humans use to justify what they perceive but the experiments from chapter 4 show that it is not always the best option, and very often implies expensive computations. Comprehension is the main bottom-up process.

Segmentation can be guided by bottom-up and top-down information. Without prior information, segmentation is performed through the evaluation of different constraints about the elements of the image. When too many segments are being generated, these constraints can be relaxed so that more elements are integrated in the same segment. On the other hand when information is available, segmentation can be guided by it. For example if we are searching for red items, the constraint for segmentation should consider this top-down guidance. An image can be segmented in many different ways, that is why top-down information is so important. A classic example in psychology works is image 5.1.

Most people without information about the image content will not be able to segment properly the image. There is no apparent relation between the black spots. However once that this information is given, human vision quickly manages to find the right segmentation criterion [1]. One of the challenges in computer vision is therefore to implement mechanisms to optimize the heuristic search.

5.2.2 The whole and its parts

Segmentation is more than a pre-processing technique, it is a relation between the parts and the whole. This relation can be useful from both sides. The parts of the whole can be reasoned as high level features to reach a conclusion about the whole. Otherwise, information about the whole can be used to categorize the parts. For example considering the parts of the the picture "La Gare de Saint Lazare" (2.7) without the whole makes their categorization more difficult,

[1] A dalmatian under a tree

Figure 5.1: Image by R.C. James

if possible at all. Another example is found in handwriting segmentation, a connected component is easier to segment in characters when we have previously categorized the connected component as a particular word (Koerich *et al.* [2005]).

Categorizing the whole by comprehension of its parts implies segmentation, recognition of the parts and comprehension, which requires more resources than the direct categorization of the whole. That is why a perceptual system should learn to directly categorize new wholes. In section 4.1 we show how activities that at first would be categorized by tracking each person, could be learned and categorized directly with a simple motion detection, and without recognizing any of its parts.

In fact the fundamental segmentation is the division between background and foreground. Objects can only be categorized when the system is able to differentiate them from background. This is also true for human vision, and a famous example was given by Rubin [1958] with the image 2.6. Depending on the segmentation criterion, a vase or two faces can be recognized, but not both at the same time.

5.2.3 Adaptation

> There is little doubt that people are not born with but acquire skills. The fundamental aspect of a skill is that an action executed now depends for its accomplishment on the execution of prior actions: activities are carried out on the basis of things done before. A machine's actions, however, are largely independent of what has gone before; its ability to do something is due to its design, not to its past experience. People acquire skills, but machines have their skills built-in (Beck *et al.* [1981]).

The limitations noted in this early work about human and machine vision define some of the requirements that computer vision needs to emulate human vision. Machines have to be able to adapt. The approach described in section 3.2.2.4 is a procedure by which the system evolves. On one side the collected information is integrated, such that new categorizations depend on the previously categorized elements. This information can also be used, such that following activities are carried out on the basis of that information.

On the other side, the information gathered improves and expands the recognition capacity of the system. The results of categorizations are useful for optimizing the parameters of the recognizers or for creating new recognizers able to directly recognize an object that has been categorized by comprehension of its parts. Information is not simply stored, it is integrated, so that the machine improves or acquires new skills. With every information gathered the perceptual system changes, evolves, adapts to be more efficient.

The results of human vision cannot be achieved with a single action, such an achievement requires a process of artificial intelligence involving knowledge representation, probabilistic reasoning, heuristic search and pattern recognition.

5.3 Future work

In section 3.3 we have reviewed some of the most relevant publications in the field of computer vision. In chapter 4 we have implemented methods for four different

and specific visual perception applications. However we have not proposed a general method for implementing machine visual perception systems. Some of the components seem to be almost ready. Methods inspired in biology have shown very promising results for recognizing scenes or objects (Farabet *et al.* [2013]; Karpathy & Fei-Fei [2015]; Krizhevsky *et al.* [2012]; Pinheiro & Collobert [2014]; Simonyan & Zisserman [2015]; Zheng *et al.* [2015]). Artificial neural networks have proved to be useful models to represent knowledge and recognize patterns. Recent research show how given a set of stimuli, unlabeled images in our case, ANN are able to learn how to recognize human faces (Le *et al.* [2013]) without need of engineered features.

Future work could explore how to implement a general method that integrates the different kinds of knowledge such that heuristic search is guided by this integration. We have seen that features and parameters used in recognition can be learned from examples, avoiding handcrafted rules. Constructing heuristic functions by learning from experience would follow the same principle, and could be a way of building systems able to adapt to different computer vision applications. The main challenge for machine visual perception is likely the construction of heuristic functions with a performance comparable to the ones implemented in the human visual system.

Appendix A

Introducción y conclusiones de la Tesis (Castellano)

A.1 Introducción

El mundo visual es probablemente la principal fuente de información para los seres humanos. Lo usamos para movernos, encontrar comida o amigos, para evitar peligros o simplemente para aprender cosas nuevas. Las representaciones del mundo visual, las pinturas, las imágenes y más recientemente el vídeo son una parte importante de nuestras vidas. La gente disfruta pintando, coleccionando arte, visitando museos, tomando fotografías o yendo al cine. Hemos desarrollado con éxito herramientas para introducir estas representaciones en los ordenadores, que ahora pueden almacenarlas fácilmente, mostrarlas o transmitirlas. Los ordenadores son de hecho una de las principales herramientas para crear o editar imágenes, lo que se conoce como gráficos por ordenador.

Muchas películas incluyen gráficos por ordenador que logran resultados impresionantes, que serían difíciles de alcanzar sin el apoyo de las máquinas. Por otro lado, algo que las personas hacemos normalmente sin esfuerzo, entender lo que representan las imágenes, sigue siendo un gran desafío para las máquinas. La comprensión del contenido de las imágenes es fundamental para implementar

sistemas de recuperación de imágenes, automatizar tareas de vigilancia o desarrollar agentes inteligentes como automóviles autodirigidos. Al campo de investigación relacionado con la comprensión de la imagen se le conoce por diferentes nombres, *computer vison*, *machine vision* o *machine visual perception*. Algunos autores diferencian entre *computer vision* y *machine vision* (Davies [2008] p.13) pero la diferencia es cuestionable. Para nosotros la única diferencia estaría en la adquisición de imágenes, *computer vision* sólo se ocupa de imágenes digitales, mientras que *machine vision* incluye las técnicas de digitalización. A menos que se indique lo contrario, en esta tesis los usaremos indistintamente. De hecho, en muchos casos preferimos usar *machine visual perception* porque incluye la palabra percepción en lugar de la visión. El significado es el mismo, pero la visión suele asociarse a los sensores, y nosotros vamos a explorar las diferencias entre los sensores y los sistemas perceptivos. En castellano los términos más habituales son visión por ordenador y visión artificial. El segundo nos parece más adecuado, especialmente si se quiere comparar con la visión humana. Nadie diría "visión por humano", en cambio comparar la visión humana con visión artificial resulta más normal. También se podría considerar el término visión automática, la visión de un autómata. De hecho la traducción de *machine learning* suele ser aprendizaje automático, y no aprendizaje por máquina. En este texto usaremos visión artificial para traducir *computer vision*, visión automática para traducir *machine vision* y percepción visual automática para *machine visual perception*.

La percepción visual automática suele estar relacionada con inteligencia artificial (AI), y esta disertación busca entender mejor esta relación. A primera vista, el concepto de IA puede parecer fácil de entender, la AI pretende conseguir que las máquinas emulen la inteligencia humana. El problema es que, aunque la inteligencia humana es algo familiar para la mayoría de las personas, la respuesta a la pregunta más sencilla *¿ Qué es la inteligencia ?* no es tan simple. En vez de considerar la naturaleza de la inteligencia, muy a menudo sólo evaluamos la inteligencia de una máquina comparándola con las acciones humanas equivalentes. Una máquina que juega al ajedrez es probablemente considerada una máquina inteligente, mientras que una que corta piezas de plástico no lo es. De hecho, la evaluación de la inteligencia de una máquina puede cambiar con el tiempo, por ejemplo, los lectores de caracteres ópticos solían ser considerados como pro-

gramas de AI en sus comienzos, pero cuando los resultados alcanzaron suficiente fiabilidad perdieron el estatus de inteligente (Schank [1991]).

El ejemplo de los lectores de caracteres ópticos puede ser extrapolado a muchas aplicaciones de visión artificial, si una máquina es inteligente o no se determina por lo impresionante que es la actividad realizada por la máquina. En esta tesis sugerimos que cualquier máquina capaz de percibir podría ser considerada una máquina inteligente cuando la percepción se entiende como un proceso de recogida de información. Esta Tesis doctoral explora los fundamentos de la visión para entender cómo los sistemas de percepción visual pueden ser construidos para emular o mejorar los resultados dados por la visión humana.

A.1.1 Vision artificial

A.1.1.1 Aplicaciones

La visión artificial está atrayendo mucho interés. Hoy es probablemente el campo de investigación más activo dentro de la inteligencia artificial. Tal vez el principal desencadenante de una actividad tan agitada es la evolución del hardware, que permite trabajar con imágenes de una manera que nunca antes se había visto. Ordenadores asequibles pueden almacenar cantidades masivas de imágenes y videos. La resolución de las cámaras digitales se mide en millones de píxeles. Incluso las CPUs de baja potencia ahora son capaces de reproducir video de alta calidad en dispositivos móviles. Millones de imágenes son tomadas y subidas todos los días. El video está en todas partes. Esa cantidad de información visual no puede permanecer inaccesible a los ordenadores, sino que debe ser explotada.

La visión artificial tiene aplicaciones en diferentes campos, como análisis de documentos (Cermeño *et al.* [2014a]; He & Schomaker [2015]; LeCun *et al.* [1989]), video vigilancia (Buch *et al.* [2011]; Cermeño *et al.* [2017b]; Hu *et al.* [2004]), evaluación de la calidad de alimentos (Sun [2016]), análisis deportivo (Moeslund *et al.* [2015]) o computación afectiva (Perez *et al.* [2014]; Picard [2000]). Estas aplicaciones ya forman parte de nuestro día a día en forma de productos como

lectores de tarjetas [1], consolas de juegos [2] or vehículos autodirigidos [3].

Los prodcutos anteriores incorporan sistemas de visión artificial que básicamente cumplen alguna de las funciones siguientes:

- Detección de objetos
- Seguimiento de objectos
- Análisis de escenas
- Clasificación de escenas

Los métodos de detección de objetos buscan encontrar objetos conocidos dentro de una imagen, mientras que en una secuencia de vídeo los métodos de seguimiento relacionan los objetos de un fotograma con los objetos de los fotogramas anteriores. Los métodos de análisis de escenas están estrechamente relacionados con la detección de objetos, sin embargo, estos últimos sólo buscan un conjunto de objetos conocidos en la imagen mientras que los primeros tratan de dividir la imagen en regiones asociadas con categorías semánticas como persona, coche, cielo, hierba etc. La detección de objetos informa de la posición del objeto cuando se encuentra mientras que el análisis de escena informa de una descripción de la escena. Finalmente, los métodos de clasificación de escenas asignan una etiqueta a una imagen o secuencia de vídeo. Pero ¿cómo podría una máquina cumplir con tales funciones?

A.1.1.2 Técnicas

Las respuestas a la pregunta anterior normalmente se encuentran en la literatura de *Digital Image Processing* y *Pattern Recognition*. La distancia entre el procesamiento de imágenes y la visión artificial no está clara. Algunos de los libros más citados en este campo se basan de algún modo o consideran útil un paradigma por el cual el procesado general se divide en tres tipos: de bajo nivel (temprano), de nivel medio (intermedio) y de alto nivel (Davies [2008]; Forsyth & Ponce [2003];

[1] www.abbyy.com
[2] www.xbox.com/es-ES/xbox-one/accessories/kinect
[3] www.tesla.com

Gonzalez & Woods [2008]). La visión de bajo nivel se ocupa de transformaciones de imagen, como filtros de eliminación de ruido u operaciones morfológicas como erosión o dilatación, y extracción de características, como detección de bordes o análisis de textura. La visión de nivel medio se ocupa de extraer información sobre las imágenes, por ejemplo formas o movimiento. La visión de alto nivel implica el reconocimiento de patrones, estableciendo una relación entre las características de la imagen y las características de objetos o escenas.

La nomenclatura del paradigma sugiere la idea del procesamiento secuencial: primero bajo nivel, luego nivel medio y finalmente procesamiento de alto nivel. La figura 1.2 reproduce un diagrama de Szeliski [2010] describiendo la relación entre diferentes técnicas de visión artificial. También sugiere un procesamiento secuencial, segmentación y detección de características conectadas de un lado al procesamiento de imágenes y de otro lado al reconocimiento, algo que podría verse como una división en tres niveles. Sin embargo, el autor advierte que "esta taxonomía debe tomarse con prudencia, ya que el procesamiento y las dependencias en este diagrama no son estrictamente secuenciales" (p.19).

A.1.2 Motivación de la Tesis

Eduardo Cermeño ha trabajado en una empresa especializada en aplicaciones de visión artificial desde 2004. Cada día, personas y empresas muestran interés en automatizar una amplia gama de tareas, desde verificación de calidad hasta análisis del comportamiento. Por ejemplo, las empresas desean saber cuántas personas entran en sus tiendas, cuáles son las áreas más visitadas, cuánto tiempo tienen que esperar los clientes antes de pagar, incluso su estado de ánimo al salir de la tienda. Los observadores humanos podrían reunir información para responder a todas estas preguntas, pero ¿podría hacerlo una máquina? En esta disertación tratamos las cuestiones fundamentales que deben ser resueltas para entender cómo las máquinas podrían emular o mejorar los resultados de la percepción visual humana: ¿qué se percibe ?, ¿cómo lo percibe una máquina? y ¿cómo construimos tal máquina?

Acotamos la primera pregunta al considerarla equivalente a ¿podría una máquina percibir todo lo que es percibido por un ser humano? La respuesta a esta pre-

gunta requiere conocimiento acerca de lo que los humanos son capaces de ver. Para tratar esta última cuestión debemos proponer una estrategia para percibir aquello que se haya respondido en la primera pregunta, y explicar por qué esta estrategia es apropiada.

La literatura relacionada con visión artificial presenta muchas técnicas, pero no explica su papel en el proceso de percepción visual. Por ejemplo, sabemos que la segmentación divide las imágenes en partes, pero ¿por qué deberíamos dividir una imagen en partes? Algunos autores consideran el reconocimiento de objetos como un proceso de alto nivel (Davies [2008]; Forsyth & Ponce [2003]), mientras que otros (Gonzalez & Woods [2008]) lo consideran un proceso intermedio, pero si estamos interesados en la clasificación de una escena ¿por qué deberíamos realizar el reconocimiento de objetos? Por paradójico que parezca, no hemos encontrado una teoría computacional explícita acerca de la percepción visual automática que explique qué se calcula y por qué. El mismo problema fue abordado por Marr [1982] para la visión humana. Su forma de enfocar la visión ha sido muy inspiradora. Las mismas preguntas y metodología utilizadas para entender la visión humana pueden ser usadas para entender mejor la visión artificial.

Marr [1982] sugiere que los hallazgos neurofisiológicos no son suficientes para entender la visión humana, la presente disertación cuestiona si la investigación en nuevas características o clasificadores es suficiente para entender cómo se podrían diseñar sistemas perceptivos comparables a la visión humana. A principios de siglo Viola & Jones [2001] y Lowe [2004] presentaron dos métodos prometedores para extraer características para el reconocimiento de objetos. En 2012, después de la publicación del conjunto de datos Imagenet (Deng *et al.* [2009]), un nuevo enfoque fue presentado por Krizhevsky *et al.* [2012], iniciando una nueva ola de métodos basados en redes neuronales convolucionales que han revolucionado el mundo del reconocimiento de objetos (Girshick *et al.* [2014]; He *et al.* [2016]; Sun & Ponce [2016]) y del análisis de escenas (Grangier *et al.* [2009]; Karpathy & Fei-Fei [2015]).

Muy a menudo la percepción visual automática se enfoca como un problema de reconocimiento de patrones. Si este fuera el caso, no estaríamos muy lejos de la solución. Simonyan & Zisserman [2015] logra una tasa de error de top-5 de 6.8 % en el desafío de reconocimiento visual Imagenet - ILSVRC- (Russakovsky

et al. [2015]). Esto significa que un 93.2 % de las imágenes tenía la etiqueta correcta entre un conjunto de 5 predicciones dadas por el algoritmo. El conjunto de pruebas ILSVC tiene 100.000 imágenes con 1000 categorías que cubren plantas, formaciones geológicas, objetos naturales, deportes, artefactos, hongos, personas, animales, comida, etc.

Sin embargo, la realidad es que no estamos tan cerca de encontrar una solución como estos resultados podrían dejar pensar. Las aplicaciones del mundo real muy a menudo van más allá del reconocimiento de objetos. Las personas son capaces de percibir aves en el cielo o en videos, incluso si apenas están representadas por unos pocos píxeles. Las personas son capaces de distinguir entre un árbol que se mueve y un intruso humano, incluso si este está disfrazado. Son capaces de reconocer los efectos de un cambio de iluminación, incluso si nunca han visto un cambio como ese antes. Esta tesis doctoral está motivada por la experiencia adquirida en una empresa que desarrolla aplicaciones de visión artificial y la voluntad de explorar cuestiones fundamentales para las que aún no se ha encontrado respuesta.

Russell & Norvig [2014] afirma que algunos de los fundadores de AI (Beal & Winston [2009]; McCarthy [2007]; Nilsson [2005]) "han expresado su descontento con el progreso de la IA". Piensan que la investigación en IA debería enfocarse menos en "versiones de aplicaciones que cada vez incluyen nuevas mejoras para tareas específicas " y " volver a sus raíces ":" máquinas que piensan, que aprenden y que crean "(p.27). Nuestra investigación se centra en máquinas que ven, en entender qué hace falta para que la percepción visual automática sea comparable con la humana. No estamos buscando nuevos métodos para resolver una tarea particular, ni un método general para implementar la percepción visual automática, estamos buscando una teoría que nos explique por qué los resultados de uno u otro sistema de visión artificial no alcanzan los ofrecidos por la visión humana.

A.1.3 La Tesis

La Tesis desarrollada en esta disertación propone un marco general teórico para explicar cuáles son los procesos requeridos para que la percepción visual automática pueda lograr los resultados de la visión humana. Podría expresarse de

la forma siguiente:

> La percepción visual automática es un proceso heurístico iterativo por el cual se reune información relacionada con una imagen. El proceso combina métodos descendientes y ascendientes para transformar un conjunto de píxeles en una jerarquía de categorías. Se procesan características de bajo nivel para reconocer lo que se ha visto antes, mientras que características de alto nivel se procesan para comprender lo que se está viendo. Un sistema de percepción visual es un agente inteligente cuyo programa se basa en tres operadores básicos: segmentación, reconocimiento y razonamiento, y cuyo objetivo es determinar si una imagen o sus partes satisfacen las condiciones de un conjunto de categorías objetivo.

A.1.4 Esquema general de la disertación

Con el fin de entender cómo una máquina podría lograr los resultados de la visión humana, el primer paso debe ser la comprensión de la naturaleza de esos resultados. Uno de los objetivos de esta tesis es analizar las principales teorías sobre la percepción visual humana. La base neurofisiológica de ésta suele estar presente en la introducción de libros sobre visión artificial, y ha inspirado varios métodos aplicados en esta área, como las Redes Neuronales Artificiales (ANNs). El estudio de las neuronas involucradas en la percepción visual muestra cómo la visión humana se implementa biológicamente pero puede no ser suficiente para entender lo que se percibe o por qué esta implementación es apropiada. La neurofisiología está estrechamente relacionada con la psicología, la rama de la ciencia que se ocupa tanto de la mente como de la percepción. Hemos revisado trabajos relevantes del campo de la psicología en busca de respuestas a preguntas como "¿ por qué las cosas se ven como se ven?" (Koffka [1935]) o "¿ por qué vemos lo que vemos ? " (Purves & Lotto [2003]). La psicología analiza los procesos de la mente detrás de la visión y explica la lógica de usar tales procesos, no sólo cómo podrían ser implementados. Marr [1982] describe estos niveles de explicación en la tabla 1.1.

El segundo objetivo es presentar un marco teórico para explicar qué cálculos son necesarios para lograr los resultados de la visión humana. La percepción visual se enfoca como una actividad de procesado de información, de la que analizamos

tanto la entrada como la salida. A partir de este análisis se propone un algoritmo con las acciones necesarias para realizar la transformación de la entrada en la salida. El marco teórico se ocupa de los niveles de explicación llamados "Teoría computacional" y "Representación y algoritmo" en la tabla 1.1. Luego revisamos varios métodos del estado del arte utilizados para implementar aplicaciones de visión artificial. Analizamos el papel que cada uno de ellos podría tener en el ámbito de nuestra teoría.

La Tesis doctoral ha sido motivada por inquietudes surgidas durante el desarrollo de aplicaciones reales. Hemos seleccionado cuatro tipos diferentes de aplicación para probar los principios de nuestra teoría. El objetivo no es encontrar el mejor método para resolver cada problema, sino evaluar si la aplicación de estos principios conduce a resultados comparables con la visión humana en una variedad de aplicaciones, y por tanto, evaluar su validez.

La Tesis se estructura en cinco capítulos, de la siguiente manera:

- El capítulo 1 presenta el tema de percepción visual, y las motivaciones, esbozos y aportes de esta tesis doctoral.

- El capítulo 2 repasa obras relacionadas con la percepción visual de los campos de la neurofisiología y la psicología, para que los resultados de la visión humana se entiendan mejor.

- El capítulo 3 presenta un nuevo marco para la percepción visual automática. Seguimos el esquema de Marr [1982] con tres niveles de explicación. Describimos primero una teoría computacional para la visión, luego una representación y un algoritmo, y finalmente revisamos los métodos más avanzados del estado del arte que podrían usarse para implementar las operaciones fundamentales del algoritmo.

- El capítulo 4 estudia cuatro aplicaciones de visión artificial con diferentes tipos de percepción: percepción de actividad, autoría, intrusión y estética. Un experto humano probablemente sugeriría enfoques basados en características de alto nivel, pero en todos los casos se pueden lograr resultados comparables a los dados por la visión humana sin seguir las sugerencias proporcionadas por una persona.

- El capítulo 5 concluye la Tesis resumiendo los principales resultados obtenidos y esbozando futuras investigaciones.

A.2 Conclusiones

A.2.1 Visión artificial y humana

A.2.1.1 ¿Puede una máquina percibir lo que percibe un ser humano?

La respuesta corta sería, sí puede. Las teorías sobre visión humana proponen que vemos formas (Marr [1982]), *affordances* (Gibson [1986]), o una representación probabilística del pasado (Purves & Lotto [2003]). Todas ellas pueden ser representadas por relaciones. Las formas pueden representarse mediante relaciones geométricas de píxeles. Las *affordances* pueden representarse por relaciones semánticas, relaciones entre un signo, una forma por ejemplo, y un concepto. Un concepto puede ser representado por la relación de sus propiedades. La percepción se basa en el estudio de relaciones, de modo que para que una máquina pueda percibir, tiene que ser capaz de manejar los diferentes tipos de relaciones involucradas en la percepción visual. En el capítulo 3 mostramos cómo cada uno de estos tipos puede ser representado y procesado por máquinas. En el capítulo 4 presentamos ejemplos de una variedad de aplicaciones de visión artificial. Los resultados de la percepción visual humana pueden emularse en campos tan diferentes como el reconocimiento de actividad, de autoría de manuscritos, de intrusión o la percepción de estética.

Cuando se relacionan varios elementos se les da forma y lo que tiene forma es etimológicamente información. La información es por tanto un elemento clave en la percepción visual. En función de la naturaleza de los elementos se definen diferentes tipos de información. Hemos propuesto agruparlos en dos tipos: información intrínseca y extrínseca. En el primer caso se trata de relaciones entre elementos intrínsecos de la imagen, píxeles, mientras que en el segundo se trata de relaciones entre elementos extrínsecos, conocimiento.

Hemos observado que los resultados de la percepción no son homogéneos, dos

personas pueden percibir el mismo objeto de manera diferente. Las imágenes 2.9 ilustran la afirmación de Purves & Lotto [2003]: "la salida de cualquier detector hacia el resto del sistema visual es necesariamente tan ambigua como el estímulo que presumiblemente codifica" (p.5). Los resultados de cualquier sistema visual, humano o artificial, no son ciertos, sino estocásticos. Cuando vemos algo en una imagen, consideramos que tenemos evidencias suficientes para afirmar que ese algo está representado en la imagen. Cuando una máquina categoriza una imagen, considera que ha encontrado evidencias suficientes para hacer tal categorización.

La categorización de una imagen depende del sistema perceptual que la categoriza, sea humano o artificial. Los resultados de distintos sistemas visuales no deberían compararse únicamente por la certeza obtenida por un clasificador de patrones, sino por la calidad de las definiciones utilizadas para categorizar. Por ejemplo, ¿representa el cuadro de Velázquez 3.1 la Corona Borealis? Es díficil responder sin evaluar la definición utilizada para realizar esta categorización. Para conseguir que una máquina emule los resultados de la visión humana necesitamos definiciones equivalentes a las usadas por los humanos para categorizar las imágenes. Los resultados de los experimentos de las secciones 4.1 y 4.4 muestran cómo una máquina puede emular la categorización de una determinada persona, incluso con mayor precisión que otra persona.

A.2.1.2 ¿Cómo puede una máquina lograr los resultados de la visión humana?

Los ordenadores son herramientas poderosas para el procesamiento de información, pueden almacenarla en cantidades masivas y realizar cálculos de forma más rápida de lo que lo podría hacer un cerebro humano. Por lo tanto, parecen adecuados para implementar sistemas de percepción visual. El desafío es encontrar los procesos adecuados que permitan realizar dicha percepción. Esta tarea supone un desafío ya que la mayor parte del conocimiento requerido en percepción visual es conocimiento por familiaridad , que no es el tipo de conocimiento que un ser humano puede expresar con el lenguaje. En inglés usamos *knowledge by acquaintance* y *propotional knowledge*. Las traducciones anteriores son las que se ajustan con mayor exactitud a los términos ingleses, sin embargo se podrían

haber usado los verbos, conocer y saber o bien conocimiento tácito y conocimiento codificado, pero esto requeriría de una discusión que va más allá de los objetivos de esta tesis.

Características de bajo nivel. A pesar de que la visión es algo natural para las personas, describir los objetos que se ven puede ser complejo, incluso para objetos simples. ¿Cómo podría una persona describir un gato de modo que pueda ser reconocido en sus diferentes poses y no confundirlo con un perro? Las descripciones dadas por los seres humanos se basan en características de alto nivel, por ejemplo, un gato tiene cuatro patas, dos orejas puntiagudas, bigote, etc. Sin embargo, las definiciones de características de alto nivel suelen ser más débiles que las definiciones basadas en características de bajo nivel, ya que las primeras están basadas en las últimas. Las características de alto nivel son sólo un subconjunto de las posibles relaciones entre las características de bajo nivel que han sido conceptualizadas, pero esto no implica que otros conjuntos de relaciones entre características de bajo nivel no sean más útiles para identificar el objeto.

Los experimentos del capítulo 4 muestran cómo los sistemas basados en características de bajo nivel son capaces de emular o mejorar los resultados dados por personas. Las características que una persona podría utilizar para describir un objeto no puede ser la mejor opción para crear una definición para un ordenador. La categorización y la descripción son acciones diferentes. Una descripción de alto nivel no es necesaria para categorizar una imagen, pero es el procedimiento común por el cual las personas transmiten conocimiento sobre reconocimiento porque es más fácil que expresar con proposiciones que características de bajo nivel.

Un problema de búsqueda. La estrategia que hemos propuesto es abordar la percepción visual como un problema de búsqueda heurística, en el que la información se recoge en cada paso para guiar el proceso de búsqueda. Al igual que cualquier otro sistema, los sistemas visuales tienen recursos limitados y en muchas situaciones del mundo real la cantidad de información disponible los sobrecargaría en gran medida. Una estrategia heurística puede producir buenos resultados con menos recursos que los enfoques basados en búsqueda exhaustiva.

En el peor de los casos, el primero puede ser equivalente al segundo, pero un sistema visual eficiente debe ser capaz de encontrar objetivos en muchas situaciones sin escanear exhaustivamente la imagen. En la visión humana, la atención visual es el mecanismo por el cual sólo se procesa una fracción de la información disponible. Además, la recopilación de información puede ser también una forma de verificación. En la sección 3.3.5.3 presentamos resultados que muestran que incluso los clasificadores de última generación pueden ser fácilmente engañados. La exploración de varios enfoques conduce a una variedad de evidencias que compensan las debilidades de los esquemas de clasificación aislados.

Agente inteligente. El proceso de búsqueda se guía por lo percibido con anteriodidad y la información conocida por el sistema. Un paradigma bien conocido en IA para estudiar los problemas de búsqueda son los agentes inteligentes. "Para cada posible secuencia de percepciones, un agente racional debe seleccionar una acción de la que se espera que maximice una medida de rendimiento dado un conjunto de evidencias proporcionado por la secuencia de percepciones y cualquier conocimiento que el agente haya incorporado" (Russell & Norvig [2014] p.38). En este paradigma la percepción suele asociarse a sensores. Sin embargo, sugerimos que sólo al considerar la percepción como el objetivo de un agente podremos emular los resultados de la visión humana. Dada una determinada imagen, un sensor siempre genera el mismo resultado. Un sistema perceptivo construido como un agente inteligente puede percibir el mismo objeto como un arma o como alimento, por ejemplo una manzana, dependiendo de la secuencia de percepciones y su conocimiento. Si el sistema ha reconocido la acción "lanzar", la manzana se consideraría como un arma y no como alimento. Con este enfoque, el resultado de la percepción visual no es una etiqueta, un término sin significado, sino la evolución de un sistema perceptivo. La percepción cambia la lista de percepciones, pero también el conocimiento del sistema. Lo que se ha visto, se puede extraer mediante la evaluación del estado del sistema. Un sistema externo podría realizar consultas al sistema visual evaluando si su estado satisface las restricciones que definen una categoría. El mismo estado se puede consultar varias veces, con diferentes restricciones de categoría, para que los sistemas externos puedan obtener información sobre la imagen. El estado del sistema visual codificaría el equiv-

alente a la Memoria Visual de Corto Plazo (VSTM). Las categorías activadas dentro del sistema visual en cada momento también podrían estar relacionadas con sistemas externos y ser vistas como percepción instantánea. La percepción instantánea puede desencadenar una reacción fuera del sistema visual o ser ignorada, cumpliendo el papel tradicional asignado a los sensores.

Conocimiento. A diferencia de los sensores, los sistemas perceptivos aprenden y se adaptan, adquieren conocimiento. La representación del conocimiento es una de las claves que han revolucionado el campo de la visión artificial en pocos años. Deng *et al.* [2009] presenta una base de datos con millones de imágenes etiquetadas, Imagenet, que supone un almacén de conocimiento nunca antes visto. Este conocimiento se representa mediante un modelo inspirado en biología: una red neuronal convolucional (CNN). Las CNNs ya habían sido utilizadas para el reconocimiento de imágenes veinte años antes (LeCun *et al.* [1990, 1995] pero con una arquitectura de alrededor de diez mil conexiones. Para representar el conocimiento de Imagenet Krizhevsky *et al.* [2012] utiliza una arquitectura con sesenta millones de parámetros , mientras que la utilizada en Simonyan & Zisserman [2015] supera los cien millones de parámetros.

A.2.2 Un sistema activo

A.2.2.1 Procesado descendiente y ascendiente

Nuestro enfoque de la percepción visual combina procesos descendentes y ascendentes. La operación clave para esto es la segmentación, cuándo y cómo dividir una imagen. El principio es segmentar el menor número de veces posible y que los métodos de segmentación iniciales estén basados en condiciones simples que generen un número limitado de segmentos. En las secciones 4.1, 4.2, 4.3 se presentan ejemplos en los que esquemas de segmentación simples logran buenos resultados, incluso cuando los resultados de la segmentación no son los que un ser humano seleccionaría.

Antes de segmentar una imagen o una región, el sistema debe extraer la mayor cantidad de información posible. La categorización directa se realiza mediante

el cálculo de características globales y locales. Las características globales representan estadísticas sobre todos los elementos de la imagen mientras que las características locales representan relaciones entre elementos vecinos sin necesidad de segmentación. La información obtenida de la categorización directa es útil para decidir si vale la pena o no la división de la imagen o región en regiones o subregiones. Este proceso es el componente descendiente del algoritmo propuesto para la percepción visual.

Por otra parte, los resultados de categorizar las partes se pueden utilizar para categorizar el todo, en un proceso llamado comprensión. Esta es probablemente la primera opción que los humanos usan para justificar lo que perciben, pero los experimentos del capítulo 4 muestran que no siempre es la mejor, ya que a menudo implica un procesado costoso. La comprensión es el principal proceso ascendente.

La segmentación puede ser guiada por información ascendente o descendente. Sin información previa, la segmentación se realiza a través de la evaluación de diferentes condiciones sobre los elementos de la imagen. Cuando se generan demasiados segmentos, estas restricciones se pueden relajar de modo que se integren más elementos en el mismo segmento. Por otro lado, cuando la información está disponible, la segmentación puede guiarse por ella. Por ejemplo, si estamos buscando elementos en rojo, la restricción de segmentación debería considerar esta orientación descendiente. Una imagen puede segmentarse de muchas maneras diferentes, por eso es tan importante la información descendiente. Un ejemplo clásico en trabajos de psicología es la image 5.1.

La mayoría de las personas sin información sobre el contenido de la imagen no podrán segmentar correctamente la imagen. No hay relación aparente entre las manchas negras. Sin embargo, una vez que esta información se da, la visión humana rápidamente logra encontrar el criterio de segmentación correcto. Uno de los retos en visión artificial es por lo tanto, implementar mecanismos para optimizar la búsqueda heurística.

A.2.2.2 El todo y las partes

La segmentación es más que una técnica de pre-procesamiento, es una relación entre las partes y el todo. Esta relación puede ser útil desde ambos lados. Las partes del conjunto pueden ser razonadas como características de alto nivel para llegar a una conclusión sobre el todo. Por otro lado, la información sobre el conjunto se puede utilizar para categorizar las partes. Por ejemplo, teniendo en cuenta las partes de la imagen "La Gare de Saint Lazare" (2.7) sin el todo su clasificación es difícil, si acaso posible. Encontramos otro ejemplo en la segmentación de manuscritos, una componente conexa es más fácil de segmentar en caracteres cuando previamente hemos categorizado la misma como una determinada palabra (Koerich *et al.* [2005]).

La categorización del todo por la comprensión de sus partes implica la segmentación, el reconocimiento y la comprensión de las partes, que requiere más recursos que la categorización directa del todo. Es por eso que un sistema perceptual debe aprender a categorizar directamente nuevos conjuntos. En la sección 4.1 mostramos cómo las actividades que en principio se categorizarían siguiendo las indicaciones dadas por una persona, podrían aprenderse y categorizarse directamente con una simple detección de movimiento, y sin reconocer ninguna de sus partes.

De hecho, la segmentación fundamental es la división entre el fondo y la figura. Los objetos sólo se pueden categorizar cuando el sistema es capaz de diferenciarlos del fondo. Esto también es cierto para la visión humana, encontramos un conocido ejemplo en la imagen 2.6. Dependiendo del criterio de segmentación, se puede reconocer un vaso o dos caras, pero no ambos al mismo tiempo.

A.2.2.3 Adaptación

> No hay duda de que la gente no nace con habilidades pero las adquiere. El aspecto fundamental de una habilidad es que la ejecución actual de una acción depende de las ejecuciones previas: las actividades se llevan a cabo sobre la base de las cosas hechas con anterioridad. Las acciones de una máquina, sin embargo, son en gran parte independientes de lo que ha pasado antes; su capacidad de hacer algo se debe a su diseño,

no a su experiencia pasada. La gente adquiere habilidades, pero las máquinas tienen sus habilidades incorporadas (Beck *et al.* [1981]).

Las limitaciones señaladas en este temprano trabajo sobre visión artificial y la humana definen algunos de los requisitos que la visión artificial necesita para emular a la visión humana. Las máquinas deben ser capaces de adaptarse. El enfoque descrito en la sección 3.2.2.4 es un procedimiento por el cual el sistema evoluciona. Por un lado, la información recogida está integrada, de modo que las nuevas categorizaciones dependen de los elementos previamente categorizados. Esta información también puede utilizarse de modo que las siguientes actividades se lleven a cabo sobre la base de esa información.

Por otro lado, la información recopilada mejora y amplía la capacidad de reconocimiento del sistema. Los resultados de las categorizaciones son útiles para optimizar los parámetros de los reconocedores o para crear nuevos reconocedores capaces de reconocer directamente un objeto que ha sido categorizado por la comprensión de sus partes. La información no se almacena sin más, se integra para que la máquina mejore o adquiera nuevas habilidades. Con cada información recogida el sistema perceptual cambia, evoluciona, se adapta para ser más eficiente.

Los resultados de la visión humana no pueden lograrse con una sola acción, semejante logro requiere de un proceso de inteligencia artificial que incluya la representación del conocimiento, razonamiento probabilístico, búsqueda heurística y reconocimiento de patrones.

A.2.3 Trabajo futuro

En la sección 3.3 hemos revisado algunas de las publicaciones más relevantes en el campo de la visión artificial. En el capítulo 4 hemos implementado métodos para cuatro aplicaciones de percepción visual diferentes y específicas. Sin embargo, no hemos propuesto un método general para implementar sistemas de percepción visual automática. Algunos de los componentes parecen estar casi listos. Métodos inspirados en biología han mostrado resultados muy prometedores para el reconocimiento de escenas u objetos (Farabet *et al.* [2013]; Karpathy &

Fei-Fei [2015]; Krizhevsky *et al.* [2012]; Pinheiro & Collobert [2014]; Simonyan & Zisserman [2015]; Zheng *et al.* [2015]). Las redes neuronales artificiales han demostrado ser modelos útiles para representar conocimiento y reconocer patrones. Investigaciones recientes muestran cómo dado un conjunto de estímulos, en nuestro caso imágenes no etiquetadas, las redes neuronales artificiales son capaces de aprender a reconocer caras humanas (Le *et al.* [2013]) sin necesidad de procesos de ingeniería de características.

El trabajo futuro podría explorar cómo implementar un método general que integre los diferentes tipos de conocimiento de tal manera que la búsqueda heurística esté guiada por dicha integración. Hemos visto que las características y los parámetros utilizados en el reconocimiento se pueden aprender de ejemplos, evitando reglas artesanales. Construir funciones heurísticas aprendiendo de la experiencia seguiría el mismo principio y podría ser una forma de construir sistemas capaces de adaptarse a diferentes aplicaciones de visión artificial. El principal desafío para la percepción visual automática es probablemente la construcción de funciones heurísticas con un rendimiento comparable al que implementa el sistema visual humano.

Appendix B

Illustrations

We include a few images that help experiencing the limitations of our vision.

B.1 Resolution

Images B.1, B.2, B.3 represent the same car that image 3.2, but isolated. This way it is easier to experience how the car fades into a sea of pixels.

Figure B.1: Low resolution image

Figure B.2: Medium resolution

Figure B.3: High resolution image

B.2 Impossible images

Figure B.4: Belvedere by M.C. Escher

Several artworks by Escher represent impossible forms. We have selected

two examples. The first one, "Belvedere" shows a plausible-looking building, which in reality is impossible. At first sight, the building may seem normal, but when observed with attention, we notice impossible structures. In the second example paradoxical information about the floor and the walls prevent us from understanding the image. How should the image be observed? When the layout of objects does not follow certain principles, the Gestalt, even something as simple as counting stairways becomes complicated.

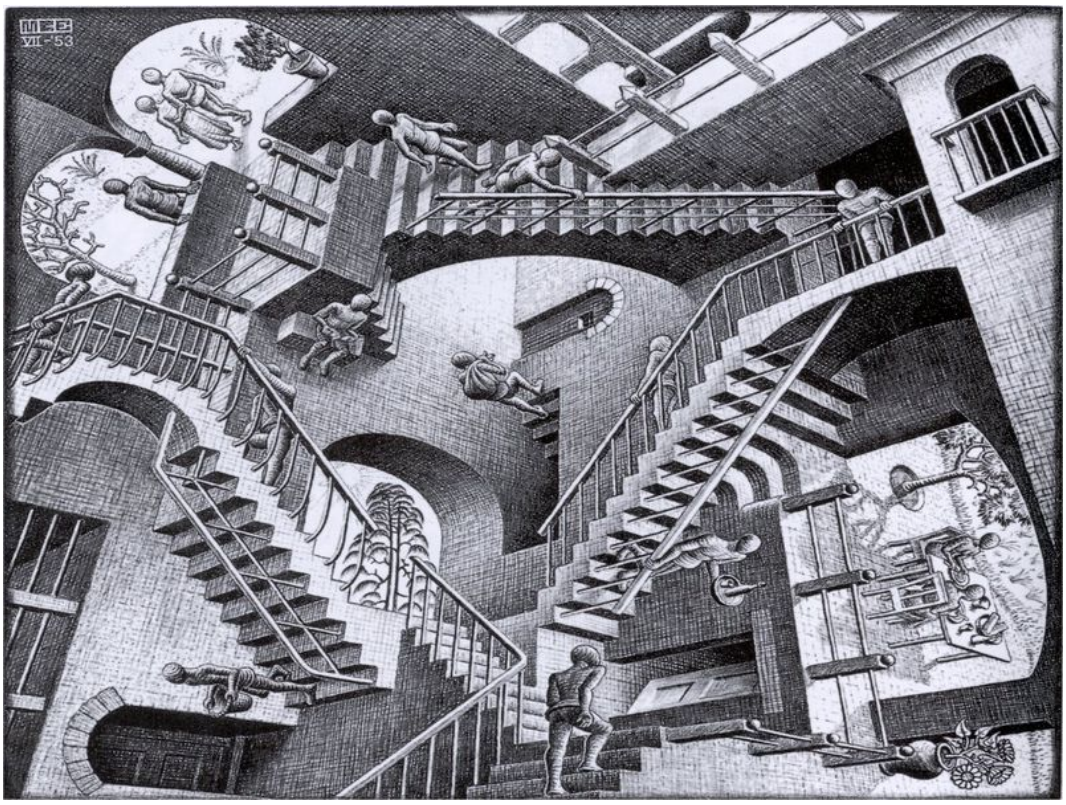


Figure B.5: Relativity by M.C. Escher

B.3 Ambiguous images

Purves & Lotto [2003] claims that the visual stimulus is ambiguous. However most of the times we have a sensation of certainty about what we see. The

following images challenge this sensation.

Figure B.6: Old and young woman - Anonymous postcard

This german postcard from 1888 shows an old woman... or maybe a young one. It is impossible to be sure without more information.

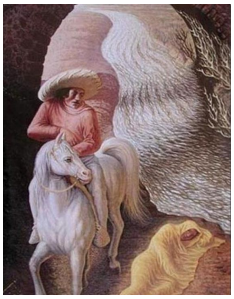


Figure B.7: Rabindranath Tagore by O. Shupliak

In image B.7 most people see an old man. It is a portrait of Rabindranath Tagore, an Indian writer, who won the Nobel prize in literature. However if we analyze the parts of the image, our perception may change. Figure B.8 reproduces the same image in a bigger scale.

Figure B.8: Rabindranath Tagore or a man riding a horse by O. Shupliak

B.4 Color

Images B.9 and B.10 are two examples of how wrong can we be about colors. The two dresses are the same color. Squares A and B are also the same color. Despite our natural confidence in color, it is not always a reliable feature for recognition.

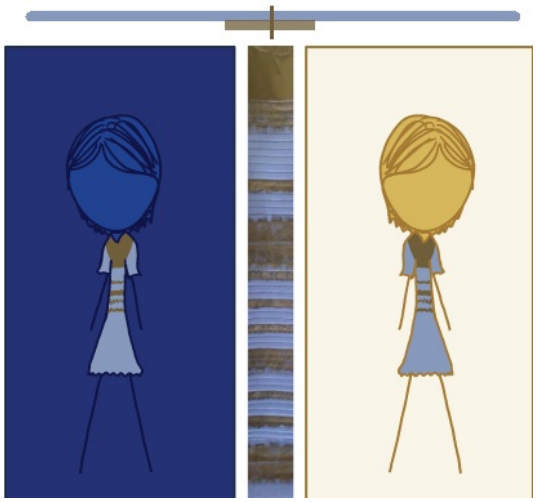


Figure B.9: Cartoon by Randall Munroe (https://xkcd.com/1492/)

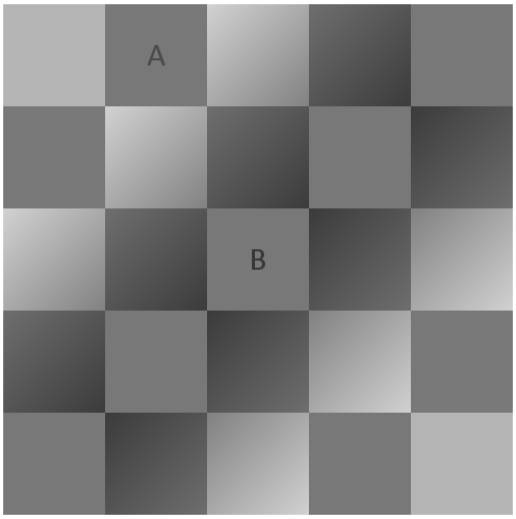


Figure B.10: The chess board illusion

B.5 Still or moving images

Image B.11 is static, however when we explore it we have the impression that the circles are moving.

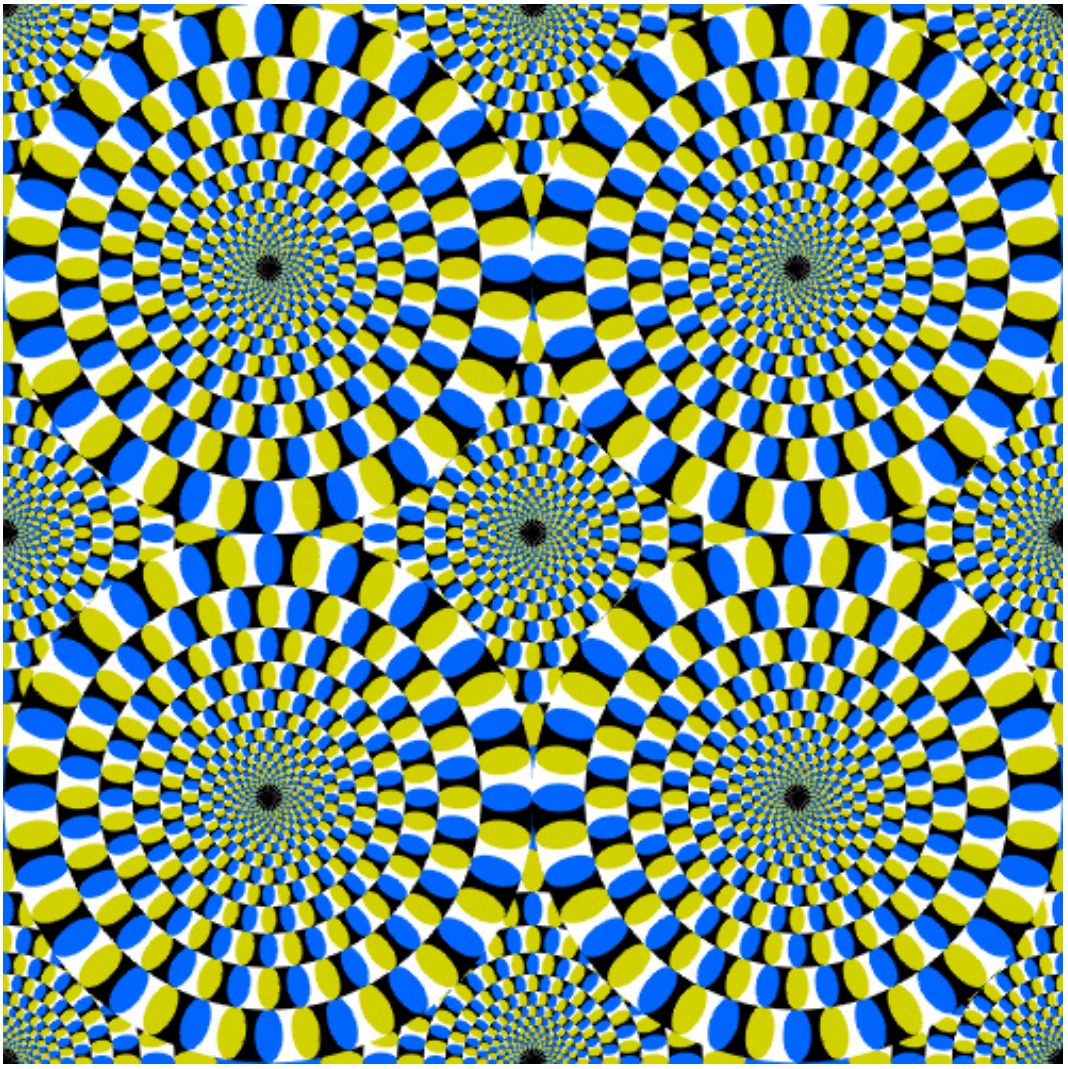


Figure B.11: Image based on “Rotating Snakes” by K. Akiyoshi

Appendix C

Experimental data

C.1 Activity perception

Class	View	Timestamp (frames)
Walking	1	14-16(0:37)&(108:163), 14-31(0:130), 14-33(0:186)
	2	14-16(0:37)&(108:168), 14-31(0:130), 14-33(0:155)
	3	14-16(0:40)&(108:165), 14-31(0:130), 14-33(0:189)
	4	14-16(0:38)&(108:172), 14-31(0:130), 14-33(0:186)
Running	1	14-16(38:107)&(164:222), 14-33(336:377)
	2	14-16(38:107)&(169:222), 14-33(336:377)
	3	14-16(41:107)&(166:222), 14-33(336:377)
	4	14-16(39:107)&(173:222), 14-33(337:377)
Splitting	1	14-31(55:130), 14-33(345:377)
	2	14-31(54:130), 14-33(344:377)
	3	14-31(51:130), 14-33(344:377)
	4	14-31(56:130), 14-33(344:377)
Merging	1	14-27(0:92)&(185:270), 14-33(106:344)
	2	14-27(0:93)&(185:271), 14-33(93:343)
	3	14-27(0:333), 14-33(94:343)
	4	14-27(0:333), 14-33(75:343)
Dispersion	1	14-27(93:133)&(271:299)
	2	14-27(94:133)&(272:300)
	3	14-27(89:138)&(268:296)
	4	14-27(96:136)
Evacuation	1	14-33(345:377)
	2	14-33(344:377)
	3	14-33(344:377)
	4	14-33(344:377)

Table C.1: Frame labelling per view

Class	Viewer	Error
Walking	1	412 frames
	2	14/412 = 3.4%
	3	95/412 = 23.06%
	4	101/412 = 24.51%
Running	1	171 frames
	2	8/171 = 4.68%
	3	8/171 = 4.68%
	4	2/171 = 1.17%
Splitting	1	109 frames
	2	9/109 = 8.26%
	3	19/109 = 17.43%
	4	17/109 = 15.6%
Merging	1	418 frames
	2	107/418 = 25.6%
	3	40/418 = 9.57%
	4	2/418 = 0.48%
Dispersion	1	70 frames
	2	15/70 = 21.43%
	3	105/70 = 150%
	4	43/70 = 61.43%
Evacuation	1	33 frames
	2	6/33 = 18.18%
	3	7/33 = 21.21%
	4	10/33 = 30.3%

Table C.2: Errors in evaluators labelling

C.2 Authorship perception

Growth i	Top 1	Top 3	Top 5	Top 10
0	26	49	58	63
1	26	49	62	70
2	36	**57**	64	76
3	**37**	54	64	76
4	28	52	63	77
6	26	44	56	77
8	36	**57**	63	75
10	34	53	62	74
12	33	49	**65**	**79**
14	36	56	63	71

Table C.3: Top-N measures of writers correctly classified (in %) with multi-segmentation shape descriptors for different growth levels. Growth i includes all the COCOs from levels $[0 \dots i]$

Growth i	LPQ + New method	LBP + New method	LPQ baseline	LBP baseline
0	86	39	84	30
1	89	37	84	30
2	89	45	84	30
3	91	51	84	30
4	91	48	84	30
6	**92**	42	84	30
8	**92**	54	84	30
10	90	48	84	30
12	91	49	84	30
14	91	53	84	30

Table C.4: Top-1 measure of writers correctly classified (in %) for different growth levels. Growth i includes all the COCOs from levels $[0 \dots i]$

Growth i	LPQ + New method	LBP + New method	LPQ baseline	LBP baseline
0	91	62	92	46
1	93	64	92	46
2	94	70	92	46
3	95	69	92	46
4	**96**	67	92	46
6	**96**	68	92	46
8	95	73	92	46
10	95	70	92	46
12	94	76	92	46
14	94	72	92	46

Table C.5: Top-3 measure of writers correctly classified (in %) for different growth levels. Growth i includes all the COCOs from levels $[0 \dots i]$

Growth i	LPQ + New method	LBP + New method	LPQ baseline	LBP baseline
0	95	66	93	63
1	95	73	93	63
2	95	80	93	63
3	**97**	81	93	63
4	**97**	82	93	63
6	96	81	93	63
8	95	84	93	63
10	**97**	80	93	63
12	95	83	93	63
14	**97**	82	93	63

Table C.6: Top-5 measure of writers correctly classified (in %) for different growth levels. Growth i includes all the COCOs from levels $[0 \dots i]$

Growth i	LPQ + New method	LBP + New method	LPQ baseline	LBP baseline
0	96	81	95	80
1	97	86	95	80
2	97	80	95	80
3	97	90	95	80
4	97	92	95	80
6	**98**	92	95	80
8	97	95	95	80
10	97	93	95	80
12	97	93	95	80
14	97	91	95	80

Table C.7: Top-10 measure of writers correctly classified (in %) for different growth levels. Growth i includes all the COCOs from levels $[0 \dots i]$

Acronyms

AGMM Adaptative Gaussian Mixture Model.

AI Artificial Intelligence.

ANN Artificial Neural Net.

BS Background Subtraction.

CCD Charge-Coupled Device.

CCTV Closed Circuit Television.

CMOS Complementary Metal Oxide Semiconductor.

CMYK Cyan Magenta Yellow and Key.

CNN Convolutional Neural Network.

COCO Connected Component.

CPU Central Processing Unit.

CRF Conditional Random Fields.

DAG Directed Acyclic Graph.

DPI Dots Per Inch.

DVR Digital Video Recorder.

ED Euclidean Distance.

ESS Efficient Subwindow Search.

FIRM Fixed Capacity Independent Race Model.

FIT Feature Integration Theory.

FOL Firt Order Logic.

FPS Frames Per Second.

FV Feature Vector.

GS Guided Search.

HSV Hue Saturation Value.

IDS Intruder Detection System.

ILSVRC Imagenet Large Scale Visual Recognition Challenge.

KB Knowledge Base.

KNN K-Nearest Neighbors.

LBP Local Binary Pattern.

LGN Lateral Geniculate Nucleus.

LPQ Local phase quantization.

MLP Multi Layer Perceptron.

OWL Ontology Web Language.

PAC Probably Approximately Correct.

PCA Principal Components Analysis.

PCC Probability of Correct Classification.

PEAS Performance Environment Actuators Sensors.

PETS Performance Evaluation of Tracking and Surveillance.

PSD Problematic Scene Detector.

RGB Red Green Blue.

RNN Recurrent Neural Network.

SIFT Scale-Invariant Feature Transform.

SOM Self Organizing Map.

SVM Support Vector Machine.

TVA Theory of Visual Attention.

VLAD Vector of Locally Aggregated Descriptors.

VSTM Visual Short Term Memory.

wff Well formed formulas.

WFV Writer Feature Vector.

YUV Luminance (Y), blueluminance (U), redluminance (V).

Glossary

Affordance What the environment offers the animal, what it provides or furnishes.

Categorize To recognize the relation between a set of elements. The process by which a set of elements are evaluated to determine whether it satisfies the constraints of a category.

Category A set of objects that satisfy the definition of the category, which is a set of constraints.

Characteristic A property that serves to identify an object.

Comprehend To take together, to unite. The process by which a set of parts is integrated into a whole.

Computational Theory The definition of *what* is computed and the reasons that explain the result of the computation (*why*).

Concept The internal representation of the knowledge about an object.

Divide To segment a whole into parts. The process by which a whole is segmented into parts.

Feature A distinctive attribute of something.

Information What has a form. Any relation of elements.

Knowledge A collection of information. In English, the term "Knowledge" is used for both knowledge by acquaintance and propositional knowledge. In other languages different terms are used, for example in Spanish: "*Conocer*" and "*Saber*", in German: "*Kennen*" and "*Wissen*".

Knowledge by acquaintance Knowledge of the relations of the impressions on the scenes which are not capable of direct verbal expressions.

Object Anything that can be a subject or a predicate, either concrete, abstract, real or fictional.

Ontology A definition of categories, properties and their relations.

Perception Information gathering.

Perceptual system A system that goes into activity in the presence of data. Its activity is to gather information starting from data. In computer vision, these data are the pixels of an image.

Property An attribute common to all members of a category. Properties are features.

Propositional knowledge Knowledge that can be expressed with propositions.

Rational agent Something that acts and whose actions are selected in order to reach a goal.

Recognizer A program that extracts and classifies features of an image or region. A recognizer represents the definition of a category.

Representation A formal system for making explicit certain types of information.

Segmenter A program that divides an image into regions. A segmenter is characterized by the constraint used to create the different regions or segments.

Signification A concept that has been related to a sign. A sign can be any set of pixels for which an inner relation has been established.

Taxonomy A classification or arrangement of categories, which has usually a hierarchical structure.

Term A sign to refer to a concept, but different to the concept and its definition.

References

Ackoff, R.L. (1989). From data to wisdom. *Journal of applied systems analysis*, **16**, 3–9. 59

Ahonen, T., Hadid, A. & Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, **28**, 2037–2041. 100

Albusac, J., Vallejo, D., Jimenez-Linares, L., Castro-Schez, J.J. & Rodriguez-Benitez, L. (2009). Intelligent surveillance based on normality analysis to detect abnormal behaviors. *International Journal of Pattern Recognition and Artificial Intelligence*, **23**, 1223–1244. 146

Alexe, B., Deselaers, T. & Ferrari, V. (2010). What is an object? *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 73–80. 107

Alexe, B., Deselaers, T. & Ferrari, V. (2012). Measuring the objectness of image windows. *IEEE transactions on pattern analysis and machine intelligence*, **34**, 2189–2202. 92, 94

Almrabet, M.M., Zerek, A.R., Chaoui, A. & Akash, A.A. (2009). Image compression using block truncation coding. *International Journal of Sciences and Techniques of Automatic Control & Computer Engineering*, **3**, 1046–1053. 116

Arbelaez, P., Maire, M., Fowlkes, C. & Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, **33**, 898–916. 94

Arbelaez, P., Hariharan, B., Gu, C., Gupta, S., Bourdev, L. & Malik, J. (2012). Semantic segmentation using regions and parts. *IEEE Conference on Computer Vision and Pattern Recognition*, 3378–3385. 94

Austin, J.L. & Warnock, G.J. (1964). *Sense and sensibilia*, vol. 83. Oxford University Press Oxford. 75, 104

Barlow, H.B. (1953). Summation and inhibition in the frog's retina. *The Journal of Physiology*, **119**, 69–88. 23, 25

Barlow, H.B. (1969). Trigger features, adaptation and economy of impulses. *Information Processing in the Nervous System*, **IV**, 209–230. 24

Barlow, H.B. (1972). Inhibition in the eye of limulus. *The Journal of General Physiology*, **1**, 371–394. 23, 24

Barnich, O. & Droogenbroeck, M.V. (2011). Vibe: A universal background subtraction algorithm for video sequences. *IEEE Transactions on Image processing*, **20**, 1709–1724. 96, 147, 148

Barron, J.L., Fleet, D.J. & Beauchemin, S.S. (1994). Performance of optical flow techniques. *International journal of computer vision*, **12**, 43–77. 96, 141

Barwise, J. (1977). An introduction to first-order logic. *Studies in Logic and the Foundations of Mathematics*, **90**, 5–46. 103

Beal, J. & Winston, P.H. (2009). Guest editors' introduction: The new frontier of human-level artificial intelligence. *IEEE Intelligent Systems*, **24**, 21–23. 7, 185

Beauchemin, S.S. & Barron, J.L. (1995). The computation of optical flow. *ACM computing surveys (CSUR)*, **27**, 433–466. 96, 141

Beck, J., Hope, B. & Rosenfeld, A. (1981). *Human and machine vision*, vol. 8. Academic Press. 177, 195

Bellinger, G., Castro, D. & Mills, A. (2004). Data, information, knowledge, and wisdom. *Available at http://systemsthinking.org/dikw/dikw.htm*. 59

Bensefia, A., Paquet, T. & Heutte, L. (2005). A writer identification and verification system. *Pattern Recognition Letters*, **26**, 2080–2092. 124

Bernardi, R., Cakici, R., Elliott, D., Erdem, A., Erdem, E., Ikizler-Cinbis, N., Keller, F., Muscat, A. & Plank, B. (2016). Automatic description generation from images: A survey of models, datasets, and evaluation measures. *J. Artif. Intell. Res.(JAIR)*, **55**, 409–442. 105, 106

Bertolini, D., Oliveira, L.S., Justino, E. & Sabourin, R. (2013). Texture-based descriptors for writer identification and verification. *Expert Systems with Applications*, **40**, 2069–2080. 100, 123, 132, 136, 137

Born, M. & Wolf, E. (1999). *Principles of Optics*. Cambridge University Press. 67

Bouwmans, T. (2014). Traditional and recent approaches in background modeling for foreground detection: An overview. *Computer Science Review*, **11**, 31–66. 96, 140

Brachman, R.J., Levesque, H.J. & Reiter, R. (1992). *Knowledge representation*. MIT press. 104

Broadbent, D. (1971). *Decision and stress*. London: Academic Press. 49

Bronstein, A., Bronstein, M. & Kimmel, R. (2008). *Numerical Geometry of Non-Rigid Shapes*. Springer Publishing Compan. 69, 70

Bruce, V. & Green, P.R. (1990). *Visual Perception. Physiology, Psychology and Ecology*. Lawrence Erlbaum Associates. 14, 20, 30

Brutzer, S., Höferlin, B. & Heidemann, G. (2011). Evaluation of background subtraction techniques for video surveillance. *IEEE Conference on Computer Vision and Pattern Recognition*, 1937–1944. 96, 139, 142, 148

Buch, N., Velastin, S.A. & Orwell, J. (2011). A review of computer vision techniques for the analysis of urban traffic. *IEEE Transactions on Intelligent Transportation Systems*, **12**, 920–939. 3, 181

Bundesen, C. (1987). Visual attention: Race models for selection from multi-element displays. *Psychological research*, **49**, 113–121. 47

Bundesen, C. (1990). A theory of visual attention. *Psychological review*, **97**, 523–547. 48

Bundesen, C. & Habekost, T. (2008). *Principles of visual attention*. Oxford Psychology Press. 48, 49, 53, 79, 83, 112

Carrasco, M. (2011). Visual attention: The past 25 years. *Vision Research*, **51**, 1484–1525. 44

Carreira, J. & Sminchisescu, C. (2010). Constrained parametric min-cuts for automatic object segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 3241–3248. 93, 94

Cermeño, E., Mallor, S. & Sigüenza, J.A. (2013). Learning crowd behavior for event recognition. *International Workshop on Performance Evaluation of Tracking and Surveillance*, 1–5. 10, 115

Cermeño, E., Mallor, S. & Sigüenza, J.A. (2014a). Offline handwriting segmentation for writer identification. *International Symposium on Biometrics and Security Technologies*, 13–17. 3, 11, 121, 181

Cermeño, E., Perez, A. & Sigüenza, J.A. (2014b). Segmentation as a characteristic for writer identification. *Submitted to Applied Intelligence*. 12

Cermeño, E., Gil, R. & Pérez, A. (2017a). Video surveillance system based on the analysis of sequences of images generated by events. P201730169. 13

Cermeño, E., Perez, A. & Sigüenza, J.A. (2017b). Intelligent video surveillance beyond background modeling. *Submitted to Expert Systems with Applications*. 3, 12, 138, 181

Chandrashekar, G. & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, **40**, 16–28. 108

Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K. & Yuille, A.L. (2015). Semantic image segmentation with deep convolutional nets and fully connected crfs. *ICLR*. 102, 109

Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on information theory*, **2**, 113–124. 104

Christlein, V., Bernecker, D. & Angelopoulou, E. (2015). Writer identification using vlad encoded contour-zernike moments. *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, 906–910. 123

Ciresan, D.C., Meier, U., Gambardella, L.M. & Schmidhuber, J. (2011). Convolutional neural network committees for handwritten character classification. *International Conference on Document Analysis and Recognition*, 1135–1139. 122

Coren, S. & Girgus, J.S. (1978). *Seeing is deceiving: The psychology of visual illusions*. JSTOR. 50

Cover, T. & Hart, P. (1967). Nearest neighbor pattern classification. *Transactions on information theory*, **13**, 21–27. 162

Coyle, K. (2006). Mass digitization of books. *The Journal of Academic Librarianship*, **32**, 641–645. 104

Dalal, N. & Triggs, B. (2005). Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **1**, 886–893. 92, 93

Danafar, S. & Gheissari, N. (2007). Action recognition for surveillance applications using optic flow and svm. *Asian Conference on Computer Vision*, 457–466. 146

Davies, E. (2008). *Computer and machine vision: theory, algorithms, practicalities*. Academic Press. 1, 4, 6, 180, 182, 184

Deng, J., Dong, W., Socher, R., Li, L., Li, K. & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. 6, 99, 174, 184, 192

Derrode, S. & Ghorbel, F. (2001). Robust and efficient fourier–mellin transform approximations for gray-level image reconstruction and complete invariant description. *Computer vision and image understanding*, **83**, 57–78. 98, 110

Descartes, R. (1897). La géométrie. *Discours de la méthode pour bien conduire sa raison et chercher la vérité dans les sciences. Plus la Dioptrique. Les Meteores. & la Geometrie qui sont des essais de cette Methode*, 297–413. 68

Deutsch, J. & Deutsch, D. (1963). Attention: Some theoretical considerations. *Psychological review*, **70**, 80–90. 49

Diller, K. (1978). *The Language Teaching Controversy*. Newbury House Publisher. 82

Dodge, R. (1903). Five types of eye movement in the horizontal meridian plane of the field of regard. *American Journal of Physiology–Legacy Content*, **8**, 307–329. 61

Dowling, J.E. (1987). *The Retina*. Harvard University Press. 18

Duda, R., Hart, P. & Stork, D. (2012). *Pattern classification*. John Wiley & Sons. 32, 146

Elhabian, S.Y., El-Sayed, K.M. & Ahmed, S.H. (2008). Moving object detection in spatial domain using background removal techniques-state-of-art. *Recent patents on computer science*, **1**, 32–54. 147

Endres, I. & Hoiem, D. (2010). Category independent object proposals. *European Conference on Computer Vision*, 575–588. 94

Everingham, M., Gool, L.V., Williams, C., KI, W., Winn, J. & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, **88**, 303–338. 92, 109

Farabet, C., Couprie, C., Najman, L. & LeCun, Y. (2013). Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, **35**, 1915–1929. 102, 108, 109, 178, 195

Feigenbaum, E.A., McCorduck, P. & Nii, P. (1989). *The Rise of the Expert Company; How Visionary Companies Are Using Artificial Intelligence to Achieve Higher Productivity and Profits*. Vintage Books. 103

Felzenszwalb, P.F. & Huttenlocher, D.P. (2004). Efficient graph-based image segmentation. *International journal of computer vision*, **59**, 167–181. 93, 94, 107

Felzenszwalb, P.F., Girshick, R.B., McAllester, D. & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, **32**, 1627–1645. 93, 94

Forsyth, D. & Ponce, J. (2003). *Computer Vision: A Modern Approach*. Prentice Hall. 4, 6, 101, 102, 109, 182, 184

Foucault, M. (2002). *The Order of Things: An Archaeology of the human sciences*. Psychology Press. 55

Fraleigh, J., , R.A.B. & Katz, V.J. (1995). *Linear algebra*. Addison-Wesley. 68

Fruhwirth, T. & Abdennadher, S. (2003). *Essentials of Constraint Programming*. Springer. 30

García-Martín, A., Hauptmann, A. & Martínez, J.M. (2011). People detection based on appearance and motion models. *International Conference on Advanced Video and Signal-Based Surveillance*, 256–260. 141, 148

Gauss, C. (2007). *General investigations of curved surfaces unabridged*. Wexford College Press. 69

Gazzah, S. & Amara, N.B. (2008). Neural networks and support vector machines classifiers for writer identification using arabic script. *International Arab Journal Information Technology*, **5**, 92–101. 126

Gibson, J.J. (1966). *The senses considered as perceptual systems*. Boston: Houghton Mifflin. 61

Gibson, J.J. (1986). *Ecological approach to visual perception*. Lawrence Erlbaum Associates. 30, 31, 33, 43, 51, 53, 60, 61, 62, 67, 70, 75, 83, 93, 97, 101, 170, 188

Girshick, R., Donahue, J., Darrell, T. & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587. 6, 95, 184

Gonzalez, R.C. & Woods, R.E. (2008). *Digital Image Processing*. Prentice Hall. 4, 6, 62, 65, 66, 67, 68, 71, 183, 184

Gonzalez, S., Sánchez, M., Lago, L., Cermeño, E., Pérez, A. & López, R. (2017). Method and device for the detection of change in illumination for vision systems. EP Patent 2,447,912. 13

Gorodnichy, D.O. (2005). Video-based framework for face recognition in video. *The 2nd Canadian Conference on Computer and Robot Vision*, 330–338. 148

Grangier, D., Bottou, L. & Collobert, R. (2009). Deep convolutional networks for scene parsing. *ICML 2009 Deep Learning Workshop*, **3**. 6, 109, 184

Gross, C.G., Rocha-Miranda, C.E. & Bende, D.B. (1972). Visual properties of neurones in inferotemporal cortex of the macaque. *Journal of Neurophysiology*, **35**, 96–111. 25

Gruber, T.R. (1995). Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, **43**, 907–928. 74

Guyon, I., Weston, J., Barnhill, S. & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, **46**, 389–422. 165

Hager, G.D., Dewan, M. & Stewart, C.V. (2004). Multiple kernel tracking with ssd. *Computer Society Conference on Computer Vision and Pattern Recognition*, **1**, I–I. 141

Hall, E. (1979). *Computer image processing and recognition*. Elsevier. 99

Hannad, Y., Siddiqi, I. & El Kettani, M.E.Y. (2016). Writer identification using texture descriptors of handwritten fragments. *Expert Systems with Applications*, **47**, 14–22. 123, 124, 132

Haralick, R.M. (1979). Statistical and structural approaches to texture. *Proceedings of the IEEE*, **67**, 786–804. 100

Haralick, R.M., Shanmugam, K. & Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, **3**, 610–621. 100, 116, 146

Hartline, H.K., Wagner, H.G. & Ratliff, F. (1956). Inhibition in the eye of limulus. *The Journal of General Physiology*, **39**, 651–673. 18

Harzallah, H., Jurie, F. & Schmid, C. (2009). Combining efficient object localization and image classification. *IEEE 12th International Conference on Computer Vision*, 237–244. 92, 93

Havasi, C., Speer, R. & Alonso, J. (2007). Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge. *Recent advances in natural language processing*, 27–29. 105

He, K., Zhang, X., Ren, S. & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778. 6, 184

He, S. & Schomaker, L. (2015). A polar stroke descriptor for classification of historical documents. *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, 6–10. 3, 181

He, X., Zemel, R.S. & Carreira-Perpiñán, M.Á. (2004). Multiscale conditional random fields for image labeling. *Computer vision and pattern recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE computer society conference on*, **2**, II–II. 100

Helmholtz, H. (1995). *Science and Culture: Popular and Philosophical Essays*. The University of Chicago Press. 58

Herzog, G. & Wazinski, P. (1994). Visual translator: Linking perceptions and natural language descriptions. *Artificial Intelligence Review*, **8**, 175–187. 105

Hoiem, D., Efros, A.A. & Hebert, M. (2007). Recovering surface layout from an image. *International Journal of Computer Vision*, **75**, 151–172. 93

Hu, M.K. (1962). Visual pattern recognition by moment invariants. *IRE transactions on information theory*, **8**, 179–187. 100

Hu, W., Tan, T., Wang, L. & Maybank, S. (2004). A survey on visual surveillance of object motion and behaviors. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, **34**, 334–352. 3, 96, 141, 181

Hubel, D.H. & Wiesel, T.N. (1959). Receptive fields of single neurones in the cat's striate cortex". *The Journal of Physiology*, **148**, 574–591. 21

Hubel, D.H. & Wiesel, T.N. (1962). Receptive fields,binocular interaction and functional architecture in the cat's visual cortex". *The Journal of General Physiology*, **160**, 106–154. 22

Hubel, D.H. & Wiesel, T.N. (1968). Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, **195**, 215–243. xii, 21

Hubel, D.H. & Wiesel, T.N. (2005). *Brain and visual perception: the story of a 25-year collaboration*. Oxford University Press. 22

Ichikawa, J.J. & Steup, M. (2001). The analysis of knowledge. *The Stanford Encyclopedia of Philosophy*. 58

Jain, A. (2010). Data clustering: 50 years beyond k-means. *Pattern recognition letters*, **31**, 651–666. 109

Junior, J.C.S.J., Musse, S.R. & Jung, C.R. (2010). Crowd analysis using computer vision techniques. *IEEE Signal Processing Magazine*, **27**, 66–77. 115

Karpathy, A. & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3128–3137. 6, 178, 184, 195

Kim, Y.S. & Street, W.N. (2004). An intelligent system for customer targeting: a data mining approach. *Decision Support Systems*, **37**, 215–228. 96, 141

Klette, R. & Rosenfeld, A. (2004). *Digital geometry : geometric methods for digital picture analysis*. Elsevier. 69

Koerich, A.L., Sabourin, R. & Suen, C.Y. (2005). Recognition and verification of unconstrained handwritten words. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**, 1509–1522. 122, 176, 194

Koffka, K. (1935). *Principles of Gestalt Psichology*. Lund Humphries. 8, 34, 36, 37, 43, 58, 74, 186

Kohonen, T. (1990). Self-organizing map. *Prodecings of the IEEE*, **78**, 1464–1480. 109, 128

Krizhevsky, A., Sutskever, I. & Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 1097–1105. 6, 95, 99, 110, 174, 178, 184, 192, 196

Kuffler, S.W. (1953). Discharge patterns and functional organization of mammalian retina. *Journal of NeuroPhysiology*, **16**, 37–68. 18

Lafferty, J., McCallum, A., Pereira, F. *et al.* (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the eighteenth international conference on machine learning, ICML*, **1**, 282–289. 100, 102

Lampert, C.H., Blaschko, M.B. & Hofmann, T. (2008). Beyond sliding windows: Object localization by efficient subwindow search. *IEEE Conference on Computer Vision and Pattern Recognition*, 1–8. 92, 107

Lampert, C.H., Blaschko, M.B. & Hofmann, T. (2009). Efficient subwindow search: A branch and bound framework for object localization. *IEEE transactions on pattern analysis and machine intelligence*, **31**, 2129–2142. 92

Laptev, I. (2006). Improvements of object detection using boosted histograms. *BMVC*, **3**, 949–958. 92

Larson, K. (2004). The science of word recognition. *Advanced Reading Thecnology*. 122

Lassaigne, J. (1973). *Les Mnines*. Skira. 56

Lazebnik, S., Schmid, C. & Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *Computer society conference on Computer vision and pattern recognition*, **2**, 2169–2178. 100

Le, Q., Monga, R., Devin, M., Chen, K., Corrado, G., Dean, J. & Ng, A. (2013). Building high-level features using large scale unsupervised learning. *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8595–8598. 110, 178, 196

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., E.Howard, R., Hubbard, W. & Jackel, L.D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, **1**, 541–551. 3, 99, 181

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W. & Jackel, L. (1990). Handwritten digit recognition with a back-propagation network, 1989. *Neural Information Processing Systems (NIPS)*, **2**, 396–404. 174, 192

LeCun, Y., Bengio, Y., *et al.* (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, **3361**, 255–258. 109, 174, 192

LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, **86**, 2278–2324. 108

LeCun, Y., Bengio, Y. & Hinton, G. (2015). Deep learning. *Nature*, **521**, 436–444. 14, 99

Lettvin, J.Y., Maturana, H.R., Mcculloch, W.S. & Pitts, W.H. (1959). What the frog's eye tells the frog's brain. *Proceedings of the IRE*, **47**, 1940–1951. 23

Lima, G. & Raghavan, K. (2014). Categories in knowledge organization. *Knowledge organization in the 21st century*, 88–95. 57

Lipton, A.J., Fujiyoshi, H. & Patil, R.S. (1998). Moving target classification and tracking from real-time video. *Fourth IEEE Workshop on Applications of Computer Vision*, 8–14. 96, 141

Liu, H. & Singh, P. (2004). Conceptnet: a practical commonsense reasoning tool-kit. *BT technology journal*, **22**, 211–226. 105

Livingstone, M.S. & Hubel, D.H. (1987). Psychophysical evidence for separate channels for the perception of form, color, movement and depth. *Journal of NeuroScience*, **7**, 3416–3468. 20

Lowe, D.G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, **60**, 91–110. 6, 98, 184

Lu, Y. & Shridhar, M. (1996). Character segmentation in handwritten words-an overview. *Pattern recognition*, **29**, 77–96. 122

Luce, R. (1963). Detection and recognition. *Handbook of mathematical psychology*, **1**, 103–189. 48

Mack, A. & Rock, I. (1998). *Inattentional Blindness*. MIT Press. 44, 45

Makadia, A., Pavlovic, V. & Kumar, S. (2010). Baselines for image annotation. *International Journal of Computer Vision*, **90**, 88–105. 146

Marques, F. & Llach, J. (1998). Tracking of generic objects for video object generation. *International Conference on Image Processing*, 628–632. 115

Marr, D. (1982). *Vision*. MIT Press. xiv, 6, 8, 9, 23, 25, 26, 27, 28, 29, 30, 43, 51, 52, 53, 64, 67, 170, 184, 186, 187, 188

Marti, U. & Bunke, H. (1999). A full english sentence database for off-line handwriting recognition. *Proceedings of the Fifth International Conference on Document Analysis and Recognition, ICDAR'99*, 705–708. 163

Marti, U. & Bunke, H. (2002). The iam-database: An english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, **5**, 39–46. 131, 163

McCarthy, J. (2007). From here to human-level ai. *Artificial Intelligence*, **171**, 1174–1182. 7, 185

McGuinness, D.L., Harmelen, F.V., Dean, M., Schreiber, G., Bechhofer, S., Hendler, J., Horrocks, I., Patel-Schneider, P.F. & Stein, L.A. (2004). Owl web ontology language overview. *W3C recommendation*, **10**, 2004. 104

Moeslund, T.B., Thomas, G. & Hilton, A. (2015). *Computer vision in sports*. Springer. 3, 181

Mohammed, D. & Abou-Chadi, F. (2011). Image compression using block truncation coding. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications*. 146

Moustapha, H. & Krishnamurti, R. (2001). Arabic calligraphy: A computational exploration. *Mathematics and design*, **1**. 159

Murphy, K., Torralba, A., Freeman, W. *et al.* (2003). Using the forest to see the trees: a graphical model relating features, objects and scenes. *Advances in neural information processing systems*, **16**, 1499–1506. 101

Murphy, K., Torralba, A., Eaton, D. & Freeman, W. (2006). Object detection and localization using local and global features. *Toward Category-Level Object Recognition*, 382–400. 101

Nakamura, J. (2005). *Image Sensors and Signal Processing for Digital Still Cameras*. CRC. Press. 65, 67

Navon, D. (1977). Forest before trees: The precedence of global features in visual perception. *Cognitive psychology*, **9**, 353–383. 143, 157, 158

Nguyen, A., Yosinski, J. & Clune, J. (2015). Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 427–436. xii, 110, 111

Nicolaou, A., Bagdanov, A.D., Liwicki, M. & Karatzas, D. (2015). Sparse radial sampling lbp for writer identification. *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, 716–720. 123

Nilsson, N.J. (1998). *Artificial Intelligence*. Morgan Kaufmann. 62, 65, 72, 73, 74, 79, 80, 103

Nilsson, N.J. (2005). Human-level artificial intelligence? be serious! *AI magazine*, **26**, 68. 7, 185

Nixon, M.S. & Aguado, A.S. (2012). *Feature extraction & image processing for computer vision*. Academic Press. 97, 98, 100, 101

Ogale, N.A. (2006). A survey of techniques for human detection from video. *Survey, University of Maryland*, **125**, 19. 139

Ojala, T., Pietikäinen, M. & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, **29**, 51–59. 100

Oliva, A. & Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, **42**, 145–175. 100

Oliva, A. & Torralba, A. (2006). Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, **155**, 23–36. 100

Oliva, A. & Torralba, A. (2007). The role of context in object recognition. *Trends in cognitive sciences*, **11**, 520–527. 101

Pearl, J. (2014). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann. 103, 105

Perez, A., Cermeno, E. & SigÜenza, J.A. (2014). Simulation of human opinion about calligraphy aesthetics. *2nd International Conference on Artificial Intelligence, Modelling and Simulation*, 9–14. 3, 11, 127, 158, 181

Perret, D.I., Rolls, E.T. & Caan, W. (1982). Visual neurones responsive to faces in the monkey temporal cortex. *Experimental Brain Research*, **47**, 329–342. 23, 24, 25

Peterfreund, N. (1999). Robust tracking of position and velocity with kalman snakes. *IEEE transactions on pattern analysis and machine intelligence*, **21**, 564–569. 141

Pham, B. (1999). Design for aesthetics: interactions of design variables and aesthetic properties. *Electronic Imaging'99*, 364–371. 161, 162

Pham, B. (2000). Shape aesthetic measures and their potential uses. *ICSC Symposia on Neural Computation*. 159

Picard, R.W. (2000). *Affective Computing*. MIT Press. 3, 158, 181

Piccardi, M. (2004). Background subtraction techniques: a review. *IEEE international conference on Systems, man and cybernetics*, **4**, 3099–3104. 96, 139, 155

Pinheiro, P. & Collobert, R. (2014). Recurrent convolutional neural networks for scene labeling. *ICML*, 82–90. 109, 178, 196

Podpora, M., Korbas, G. & Kawala-Janik, A. (2014). Yuv vs rgb - choosing a color space for human-machine interaction. *FedCSIS Position Papers*, 29–34. 66

Poppe, R. (2010). A survey on vision-based human action recognition. *Image and vision computing*, **28**, 976–990. 146

Potter, S. (2003). A survey of knowledge acquisition from natural language. *TMA of Knowledge Acquisition from Natural Language*. 103

Powers, D.M. (2011). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*. 103, 147

Princen, J., Illingworth, J. & Kittler, J. (1992). A formal definition of the hough transform: properties and relationships. *Journal of Mathematical Imaging and Vision*, **1**, 153–168. 98

Prinzmetal, W., Zvinyatskovskiy, A., Gutierrez, P. & Dilem, L. (2009). Voluntary and involuntary attention have different consequences: the effect of perceptual difficulty. *The quarterly journal of experimental psycology*, **62**, 352–369. 44

Purves, D. & Lotto, R.B. (2003). *Why we see what we do, an empirical theory of vision*. Sinauer Associates. 8, 20, 39, 40, 42, 83, 84, 90, 169, 170, 171, 186, 188, 189, 201

Quiroga, R.Q., Reddy, L., Kreiman, G., Koch, C. & Fried, I. (2005). Invariant visual representation by single neurons in the human brain. *Nature*, **435**, 1102–1107. 25

Quiroga, R.Q., Kreiman, G., Koch, C. & Fried, I. (2008). Invariant visual representation by single neurons in the human brain. *Trends in Cognitive Sciences*, **12**, 87–91. 25

Rankin, S., Cohen, N., Maclennan-Brown, K. & Sage, K. (2012). Cctv operator performance benchmarking. *International Carnahan Conference on Security Technology*, 325–330. 140

Ranzato, M., Huang, F., Boureau, Y.L. & Lecun, Y. (2007). Unsupervised learning of invariant feature hierarchies with applications to object recognition. *Conference on Computer Vision and Pattern Recognition, 2007. CVPR'07*, 1– 8. 110

Rehman, A. & Saba, T. (2012). Off-line cursive script recognition: current advances, comparisons and remaining problems. *Artificial Intelligence Review*, **37**, 261–288. 122

Reid, T. (1819). *Essays on the power of the human mind*. Bell. 54

Rubin, E. (1958). Figure and ground. *Readings in perception*, 194–203. 34, 94, 176

Rumelhart, D. (1970). A multicomponent theory of the perception of briefly exposed visual displays. *Journal of Mathematical Psychology*, 191–218. 49

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C. & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, **115**, 211–252. 7, 184

Russell, S. & Norvig, P. (2014). *Artificial Intelligence A Modern Approach*. Pearson. 7, 61, 62, 63, 73, 74, 75, 76, 77, 90, 103, 173, 185, 191

Saba, T., Rehman, A. & Sulong, G. (2011). Cursive script segmentation with neural confidence. *Int J Innov Comput Inf Control (IJICIC)*, **7**, 1–10. 122

Said, H.E.S., Peake, G.S., Tan, T.N. & Baker, K.D. (1998). Writer identification from non-uniformly skewed handwriting images. *BMVC*, 1–10. 136

Salari, V. & Sethi, I.K. (1990). Feature point correspondence in the presence of occlusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, 87–91. 141

Sanin, A., Sanderson, C. & Lovell, B.C. (2012). Shadow detection: A survey and comparative evaluation of recent methods. *Pattern recognition*, **45**, 1684–1695. 139

Saxena, S., Brémond, F., Thonnat, M. & Ma, R. (2008). Crowd behavior recognition for video surveillance. *International Conference on Advanced Concepts for Intelligent Vision Systems*, 970–981. 115

Schank, R.C. (1991). Where's the ai? *AI magazine*, **12**, 38. 2, 181

Schneiderman, H. & Kanade, T. (2004). Object detection using the statistics of parts. *International Journal of Computer Vision*, **56**, 151–177. 98

Schomaker, L. (2008). Writer identification and verification. *Advances in Biometrics, Springer Verlag*. 123

Schomaker, L. & Bulacu, M. (2004). Automatic writer identification using connected-component contours and edge-based features of uppercase western script. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**, 787–798. 123, 126, 128

Schomaker, L., Bulacu, M. & Franke, K. (2004). Automatic writer identification using fragmented connected-component contours. *International Workshop on Frontiers in Handwriting Recognition, 2004. IWFHR-9*, 185–190. 123, 128, 132, 133

Sherrington, C.S. (1941). *Man on his nature*. Cambridge University Press. 24

Shibuya, H. & Bundesen, C. (1988). Visual selection from multielement displays: Measuring and modeling effects of exposure duration. *Human Perception and Performance*, **14**, 591–600. 47

Simons, D.J. & Chabris, C.F. (1999). Gorillas in our midst:sustained inattentional blindness for dynamic events. *Perception*, **28**, 1059–1074. 44, 45, 140

Simonyan, K. & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *ICLR*. 7, 33, 104, 110, 174, 178, 184, 192, 196

Snowden, R., Thompson, P. & Troscianko, T. (2006). *Basic Vision and introduction to visual perception*. Oxford University Press. 14, 18, 20, 44

Sowa, J.F. (2006). Semantic networks. *Encyclopedia of Cognitive Science*. 105

Speer, R. & Havasi, C. (2012). Representing general relational knowledge in conceptnet 5. *LREC*, 3679–3686. 105

Srihari, R.K. (1994). Computational models for integrating linguistic and visual information: A survey. *Artificial Intelligence Review*, **8**, 349–369. 105

Srihari, S.N., Cha, S.H., Arora, H. & Lee, S. (2002). Individuality of handwriting. *Journal of forensic science*, **47**, 1–17. 121

Stamatopoulos, N., Gatos, B., Louloudis, G., Pal, U. & Alaei, A. (2013). Icdar 2013 handwriting segmentation contest. *International Conference on Document Analysis and Recognition*, 1402–1406. 122

Sternberg, S. (1966). High-speed scanning in memory. *Science*, **153**, 652–654. 45

Sternberg, S. (1969). Memory-scanning: mental processes revealed by reaction-time. *American Scientist*, **57**, 421–457. 45

Stockman, G.C. & Agrawala, A.K. (1977). Equivalence of hough curve detection to template matching. *Communications of the ACM*, **20**, 820–822. 98

Stone, J. (2013). *Parallel Processing in the Visual System*. Springer. 20

Studtmann, P. (2014). *Aristotle's Categories*. Edward N. Zalta. 54, 58

Sun, D.W. (2016). *Computer vision technology for food quality evaluation*. Academic Press. 3, 181

Sun, J. & Ponce, J. (2016). Learning dictionary of discriminative part detectors for image categorization and cosegmentation. *International Journal of Computer Vision*, **120**, 111–133. 6, 184

Szeliski, R. (2010). *Computer vision: algorithms and applications*. Springer Science & Business Media. xii, 4, 5, 183

Tao, J. & Tan, T. (2005). Affective computing: A review. *International Conference on Affective computing and intelligent interaction*, 981–995. 159

Tapiador, M. & Sigüenza, J.A. (2004). Writer identification method based on forensic knowledge. *Biometric Authentication*, 555–561. 121, 122, 137

Teague, M.R. (1980). Image analysis via the general theory of moments. *JOSA*, **70**, 920–930. 100, 116

Torralba, A. & Oliva, A. (2003). Statistics of natural image categories. *Network: computation in neural systems*, **14**, 391–412. 100

Torralba, A., Oliva, A., Castelhano, M.S. & Henderson, J.M. (2006). Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological review*, **113**, 766. 158

Treisman, A. (1977). Focused attention in the perception and retrieval of multidimensional stimuli. *Perception and Psychophysics*, **22**, 1–11. 46

Treisman, A. & Gelade, G. (1980). A feature integration theory of attention. *Cognitive Psychology*, **12**, 97–136. 45, 46

Tu, P., Sebastian, T., Doretto, G., Krahnstoever, N., Rittscher, J. & Yu, T. (2008). Unified crowd segmentation. *European Conference on Computer Vision*, **3**, 691–704. 115

Uijlings, J.R., Sande, K.E.V.D., Gevers, T. & Smeulders, A.W. (2013). Selective search for object recognition. *International journal of computer vision*, **104**, 154–171. 94, 95, 107, 124

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer science & business media. 108

Viola, P. & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Computer Society Conference on Computer Vision and Pattern Recognition*, **1**, I–I. 6, 98, 106, 110, 184

Viola, P. & Jones, M.J. (2004). Robust real-time face detection. *International journal of computer vision*, **57**, 137–154. 92, 106

Wallace, W. (2011). *The Elements of Philosophy: A Compendium of Philosophers and Theologians*. Wipf and Stock Publisher. 57, 58, 59, 90

Weiss, S., Achtelik, M.W., Lynen, S., Chli, M. & Siegwart, R. (2012). Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments. *IEEE International Conference on Robotics and Automation (ICRA)*, 957–964. 98

Wertheimer, M. (1938). Laws of organization in perceptual forms. *A source book of Gestalt psichology*, 71–88. 36

Wickenden, W.E. (1910). *Illumination and Photometry*. Read Books. 15

Winston, P. (1993). *Artificial Intelligence 3rd edition*. Addison-Wesley. 104

Wolfe, J.M. (1994). Guided search: An alternative to the feature integration model for visual search. *Psychonomic Bulletin & Review*, **1**, 202–238. 47

Wolfe, J.M., Cave, K. & Franzel, S. (1989). Guided search: An alternative to the feature integration model for visual search. *Journal of Experiemntal Psychology: Human Perception and Performance*, **15**, 419–433. 46

Xiao, T., Xu, Y., Yang, K., Zhang, J., Peng, Y. & Zhang, Z. (2015). The application of two-level attention models in deep convolutional neural network

for fine-grained image classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 842–850. 107

Xiong, Y.J., Wen, Y., Wang, P.S. & Lu, Y. (2015). Text-independent writer identification using sift descriptor and contour-directional feature. *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, 91–95. 123

Xu, S., Lau, F., Cheung, W., William, K. & Pan, Y. (2005). Automatic generation of artistic chinese calligraphy. *IEEE Intelligent Systems*, **20**, 32–39. 159

Xu, S., Jiang, H., Lau, F. & Pan, Y. (2007). An intelligent system for chinese calligraphy. *Proceedings Of The National Conference On Artificial Intelligence*, **22**, 1578. 159

Xu, Y., Dong, J., Zhang, B. & Xu, D. (2016). Background modeling methods in video analysis: A review and comparative evaluation. *CAAI Transactions on Intelligence Technology*, **1**, 43–60. 96, 148

Yan, J., Fan, Z., Wei, T., Qian, W., Zhan, M. & Wei, F. (2010). Fast and reversible surface redox reaction of graphene–mno 2 composites as supercapacitor electrodes. *Carbon*, **48**, 3825–3833. 141

Yang, M., Kpalma, K. & Ronsin, J. (2008). A survey of shape feature extraction techniques. *Patttern Recognition*, 43–90. 161

Yantis, S. (2001). *Visual Perception: Essential Readings*. Psychology Press. 14

Zhang, Y., Liu, Y., He, J. & Zhang, J. (2013). Recognition of calligraphy style based on global feature descriptor. *International Conference on Multimedia and Expo (ICME)*, 1–6. 159

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C. & Torr, P. (2015). Conditional random fields as recurrent neural networks. *Proceedings of the IEEE International Conference on Computer Vision*, 1529–1537. 109, 178, 196

Zhong, J. & Sclaroff, S. (2003). Segmenting foreground objects from a dynamic textured background via a robust kalman filter. *International Conference on Computer Vision*, 44–50. 141

Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. *Proceedings of the 17th International Conference on Pattern Recognition*, **2**, 28–31. 96