



Repositorio Institucional de la Universidad Autónoma de Madrid

<https://repositorio.uam.es>

Esta es la **versión de autor** de la comunicación de congreso publicada en:
This is an **author produced version** of a paper published in:

Statistical Methods in Video Processing: ECCV 2004 Workshop SMVP 2004,
Prague, Czech Republic, May 16, 2004, Revised Selected Papers. Lecture
Notes in Computer Science, Volumen 3247. Springer 2004. 165-174.

DOI: http://dx.doi.org/10.1007/978-3-540-30212-4_15

Copyright: © 2004 Springer-Verlag

El acceso a la versión del editor puede requerir la suscripción del recurso
Access to the published version may require subscription

Factorization of natural 4×4 patch distributions

Kostadin Koroutchev and José R. Dorronsoro *

Depto. de Ingeniería Informática and Instituto de Ingeniería del Conocimiento
Universidad Autónoma de Madrid, 28049 Madrid, Spain

Abstract. The lack of sufficient machine readable images makes impossible the direct computation of natural 4×4 image block statistics and one has to resort to indirect approximated methods to reduce their domain space. A natural approach to this is to collect statistics over compressed images; if the reconstruction quality is good enough, these statistics will be sufficiently representative. However, a requirement for easier statistics collection is that the method used provides a uniform representation of the compression information across all patches, something for which codebook techniques are well suited. We shall follow this approach here, using a fractal compression-inspired quantization scheme to approximate a given patch B by a triplet (D_B, μ_B, σ_B) with σ_B the patch's contrast, μ_B its brightness and D_B a codebook approximation to the mean-variance normalization $(B - \mu)/\sigma$ of B . The resulting reduction of the domain space makes feasible the computation of entropy and mutual information estimates that, in turn, suggest a factorization of the approximation of $p(B) \simeq p(D_B, \mu_B, \sigma_B)$ as $p(D_B, \mu_B, \sigma_B) \simeq p(D_B)p(\mu)p(\sigma)\Phi(\|\nabla B\|)$, with Φ being a high contrast correction.

1 Introduction

The importance of understanding the statistical behavior of natural images is plainly obvious for a wide range of topics, going from standard visual information processing tasks to the study of basic human visual behavior. However, direct statistics computation is not possible even for 4×4 natural image blocks: current lossless image compression techniques do not allow to go below 2.5 bits per pixel rates [12], which implies that the representation of 4×4 blocks will require in average about $16 \times 2.5 = 40$ bits. In other words, direct natural block statistics would require about $2^{40-16} \simeq 16 \times 10^6$ natural 1024×1024 images and, simply, there are not so many machine readable raw images. This has led many researchers to collect and analyze statistics not directly on blocks B but rather on appropriate, low dimensional transformations $T(B)$. Typically, the block transformation computed prior to statistics collection either reduces a block's dimension by selecting a few points, projecting the block pixels on some directions or computing a certain integral transform [7, 10] or, on the other hand, $T(B)$ allows a certain reconstruction of B [4, 9]. An example of this are those

* With partial support of Spain's CICYT, TIC 01-572.

wavelet transform methods that compute statistics over the first few wavelet components. This is approximately equivalent to compute statistics over lower resolutions of the original patches. A good recent review of the literature on the statistics of natural images is given by in [11]. In our context, desirable characteristics of the $T(B)$ transformation are

1. The information loss due to $T(B)$ should be as small as possible, so that it is meaningful to deduce statistical properties of B from those of $T(B)$. Clearly this does not hold for the methods in the first class above.
2. The data structure of $T(B)$ should be obviously the same for all B so that statistics are collected over a uniform data set.
3. Finally, the computation of $T(B)$ should be quite fast, if only to alleviate the computationally very demanding task of statistics collection.

The first requirement shows that methods of the second type are clearly the most appropriate ones, as they connect statistics collection with image compression, while the second requirement may penalize optimal compression schemes such as DCT JPEG or wavelets, as the resulting $T(B)$ could be highly non uniform. The natural alternative, that we shall use here, are codebook image compression methods that compress a given block B by choosing another D_B from a certain codebook $\mathcal{D} = \{D\}$ of mean and variance normalized domains such that

$$D_B = \arg \min_D \text{dist}(B, D) = \arg \min_D \{\|B - (\sigma_B D + \mu_B)\|_\infty\}. \quad (1)$$

Here $\|\cdot\|_\infty$ denotes the pixel-wise supremum norm and $\sigma_B D + \mu_B$ is a gray level transformation of D , with the block's standard deviation σ_B and mean μ_B being respectively taken as a contrast factor and a luminance shifting. B is then compressed by the triplet (D_B, σ_B, μ_B) . We shall see this triplet as the transformation of B , that is, $T(B) = (D_B, \sigma_B, \mu_B)$. Although clearly inspired by fractal image compression, we shall look at the (D_B, σ_B, μ_B) coding and its associated reconstruction

$$B \simeq \sigma_B D_B + \mu_B \quad (2)$$

from a codebook point view. In any case, the coding $T(B)$ certainly meets our second requirement and, as we shall see, also the first one, as the approximation it provides is close enough. Turning our attention to the third requirement, a fast computation of σ_B and μ_B can be easily achieved. Finding D_B , however, can very time consuming as it will involve full block comparisons. To minimize their number, we shall use here a hash based block precomparison, inspired in hash-based fractal image compression (FIC), a novel image compression method proposed by the authors [3], whose performance is comparable to other state of the art FIC methods (or even better in some instances). More precisely, we shall compute first a certain hash-like function $h(D)$ for all codebook domains, and distribute in the same linked list those D with the same h value. To code a given block B , a set $H(B)$ of small perturbations of the hash value $h(B)$ will



Fig. 1. Lena's image is a well known source of fractal codebooks, but statistics computed from other codebooks are similar, provided the source image is "rich" enough, as the one from the Van Hateren's database exemplified here. Its decimated square center has been used as an alternative domain source.

be computed and the full block comparisons in (1) will be done only between B and those D such that $h(D) \in H(B)$.

More details on this hash based block-domain matching are given in section 2, where we shall also discuss the basic statistics collection procedure for the $T(B) = (D_B, \sigma_B, \mu_B)$ approximations. In section 3 we shall compute the mutual information between the joint probability $p(D_B, \sigma_B, \mu_B)$ and the $P(D_B)$, $p(\sigma_B)$ and $p(\mu_B)$ marginal probabilities and see that, in a first approximation, we have for 4×4 natural image patches B that

$$p(D_B, \sigma_B, \mu_B) \simeq p(D_B)p(\mu_B)p(\sigma_B), \quad (3)$$

while a second order approximation is

$$p(B) \simeq p(D_B)p(\mu_B)(p(\sigma_B)\Phi(\|\nabla B\|)), \quad (4)$$

with $\Phi(\|\nabla B\|)$ a high contrast correction. For this we shall approximate about 280 million natural 4×4 patches extracted from the well known van Hateren database [2] using two different FIC codebooks, derived from the well known Lena image and from a typical van Hateren image depicted in figure 1. In section 4 we shall also analyze the structure of the $p(D)$ and $p(\sigma)$ probabilities ($p(\mu)$ can be easily manipulated and does not carry significant information). We shall show that $p(\sigma)$ (that is independent from the codebook used) has an exponential structure and that $p(D^B)$ follows for both codebooks a nearly uniform behavior with respect to volume in image space. The paper ends with some other comments and pointers to further work.

Quantity	Value(Lena)	Value(VH)	Limit estimates
N	231511046	231441592	—
$\log_2 N$	27.7865	27.7861	—
$H(i, j, s, \sigma, \mu)$	26.7141	26.6517	29.84
$H(i, j, s)$	17.8156	17.6565	17.73
$I(i, j, s \sigma, \mu)$	1.5642	1.4674	0.427
$I/H(i, j, s, \sigma, \mu)$	5.86%	5.51%	1.43%
$H(\sigma, \mu)$	10.4627	10.4626	10.46
$I(\sigma \mu)$	0.1698	0.1697	0.115
$I(\sigma, \mu)/H(\sigma, \mu)$	1.62%	1.62%	1.10%

Table 1. Different entropy measures (in bits) of image statistics using the Lena (second column) and van Hateren (third column) codebooks and limit estimates (fourth column) for them.

2 Methods

The approximation $B = (\tilde{B}, \sigma_B, \mu_B) \simeq (D_B, \sigma_B, \mu_B) = T(B)$ implies that $\tilde{B} = (B - \mu_B)/\sigma_B \simeq D_B$. This approximation must hold for all block pixels which, as we shall argue below, suggests to define a hash-like function

$$h(D) = \sum_{h=1}^H \left(\left\lfloor \frac{D_{i_h j_h}}{\lambda} \right\rfloor \% C + \frac{C}{2} \right) C^{h-1} = \sum_{h=1}^H b_h C^{h-1}. \quad (5)$$

to speed up domain searches. Here we shall take $H = 5$ and the points $D_{i_h j_h}$, $1 \leq h \leq H$ used are the four corner pixels and an extra middle pixel. C will be 16 and the modulus operator $\%C$ gives integer values between $-C/2$ and $C/2$. Finally λ is chosen so that (5) defines an approximately uniform base C expansion that speeds up hash searches. Therefore, we want the b_h to be approximately distributed between 0 and 16, which can be achieved if λ is chosen so that the $D_{i_h j_h}$ are uniformly distributed in $[-\lambda \frac{C}{2}, \lambda \frac{C}{2}]$. An optimal λ would then be about 0.2, although we shall take $\lambda = 2$ in what follows. Codebook domains will be derived from a 256×256 versions of the Lena and van Hateren images by extracting all its 4×4 (overlapping) blocks. This gives $(256 - 4 + 1)^2 \simeq 2^{16}$ codebook domains, that become 2^{20} after adding for each block its 8 isometries and its negative (notice that the dilations in (2) are positive). Flat domains, i.e., those such that $\sigma(D) \leq 4$, may give distorted values in (5) and we will exclude them (about 25% of both codebook domains).

Full block comparisons for a natural block B , that is, the computation of $\text{dist}(B, D) = \sup |B_{ij} - \sigma_B D_{ij} - \mu_B|$ in (1), over all block pixels are performed only over domains D such that $h(D) \in H(B)$, with $H(B) = \{h_\delta(B)\}$, where

$$h_\delta(B) = \sum_{h=1}^H \left(\left\lfloor \frac{B_{i_h j_h} - \mu_B}{\lambda \sigma_B} + \delta_h \right\rfloor \% C + \frac{C}{2} \right) C^{h-1} = \sum_{h=1}^H r_h^\delta C^{h-1}, \quad (6)$$

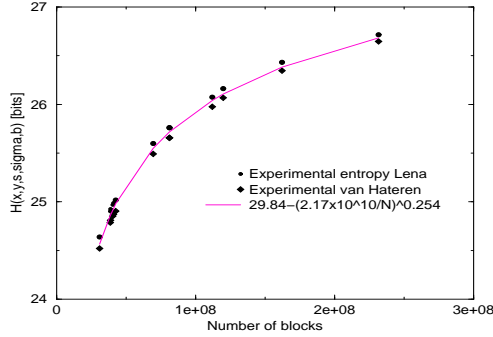


Fig. 2. Large sample behavior of the total entropy $H(i, j, s, \sigma, \mu)$ estimates. computed over Lena and van Hateren codebooks. Both values are close, but do not reach a saturation limit.

with the displacement vector $\delta = (\delta_1, \dots, \delta_H)^t$ verifying $|\delta_h| \leq 1$. Notice that the equality $h_\delta(B) = h(D)$ implies pixel closeness of the H points used to define h and, hence, a starting similarity between the D and B blocks. As a further acceleration factor, we shall content ourselves with approximate domain searches, in the sense that we will fix a tolerance value d and stop looking for domains matching a patch B as soon as a D is found such that $d(B, D) \leq d$. We shall take $d = 8$, that guarantees a reasonable reconstruction PSNR of about 30 Dbs. Finally, the coding of B will then be

$$T(B) = (i, j, s, \sigma, \mu)$$

where (i, j) indicates the position in the codebook image of the left upper corner of the matching domain, and s is an index for the isometry and negative used. As the natural patch source, we shall work with 4300 8 bit gray level images of size 1540×1024 from the van Hateren database. We shall restrict ourselves to their 1024×1024 squared centers. As done for domains, we will also exclude flat blocks, that is, those B with $\sigma \leq 4$ (about 20% of all database patches). This leaves us with a sample of about 232×10^6 natural 4×4 patches.

3 Distribution factorization

Denoting by N the number of sample patches and by M the number of domains, we should have $N \gg M$ in order to achieve accurate entropy estimates [8]. However, table 1 shows this not to be the case when estimating the full sample entropy $H_N(i, j, s, \sigma, \mu)$ of the $p(i, j, s, \sigma, \mu)$ distribution, something that can also be appreciated in figure 2, that shows that although close, the Lena and van Hateren full entropy values do not reach a saturation point. On the other hand, we have indeed $N \gg M$ when estimating the marginal entropies $H_N(i, j)$, $H_N(\sigma)$ and $H_N(\mu)$. As it can be seen in table 1, the Lena and van Hateren

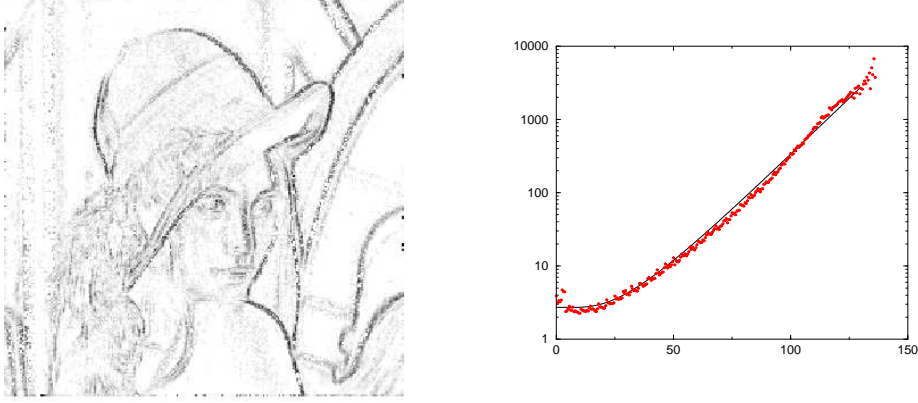


Fig. 3. Left: deviations from the first order factorization (7) arise over the edges of the codebook source image. Right: values of $E_{||\nabla B||}$ in logarithmic scale and second order epitaxy inspired approximation.

$H_N(i, j)$ entropy values are again very close ($H_N(\sigma)$ and $H_N(\mu)$ are codebook-independent). From the table one can deduce that σ and μ are independent, as their mutual information is less than 2% of their joint entropy. When looking at the dependence between the (i, j, s) and (σ, μ) distributions, the table shows that the mutual information $I(i, j, s || \sigma, \mu)$ is for both codebooks about 1.5 bits, that is, about 5.5% of the joint entropy. Although not totally independent, this points out to a first order factorization of the joint (i, j, s, σ, μ) density as the product

$$p(i, j, s, \sigma, \mu) \simeq p(i, j, s)p(\sigma)p(\mu). \quad (7)$$

In order to visualize where the remaining 1.5 bit dependence may arise, we have looked at the average $d(i, j)$ over (s, σ, μ) of the quotient

$$\frac{p(i, j, s, \sigma, \mu)}{p(i, j, s)p(\sigma)p(\mu)}.$$

The values of $d(i, j)$ can be projected over the codebook source image, where we should look for those different from one. When this is done (see figure 3, left, where $\log d(i, j)$ is depicted), it is clear that image edges are where to look in order to correct (7). This also suggests to correct (7) as

$$p(B) \simeq p(D_B)p(\mu_B)p(\sigma_B)\Phi(||\nabla B||).$$

This second approximation clearly implies that we should have

$$\Phi(||\nabla B||) \simeq E_{||\nabla B||} \left[\frac{p(i, j, s, \sigma, \mu)}{(p(i, j, s)p(\sigma)p(\mu))} \right] \quad (8)$$

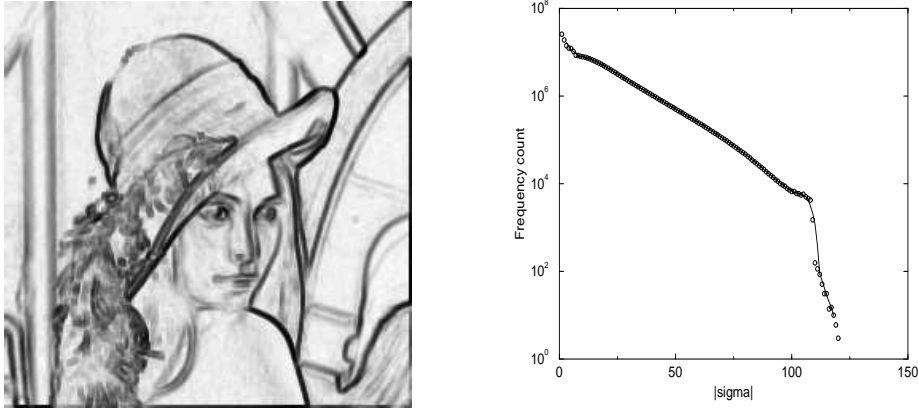


Fig. 4. Left: projection over Lena of the $-\log p(\sigma(i, j))$ distribution. Right: relative frequencies of σ in a logarithmic scale. The linear central behavior suggests an exponential distribution.

with $E_{||\nabla B||}$ denoting the conditional expectation with respect to $||\nabla B||$. This expectation is depicted in figure 3, right, which also shows that a good approximation for $\Phi(x)$ is given by

$$\Phi(x) \simeq a + b\phi\left(\frac{x}{c}\right) = a + b\left(\frac{x}{c} - \frac{\pi^2}{6} - 2\sum_{n=1}^{\infty} \frac{(-1)^n}{n^2} e^{-\frac{\pi}{c}n^2}\right),$$

where $a \simeq 2$, $c \simeq 31$ and $b/c \simeq 0.068$. The motivation for the $\phi(t)$ function comes from epitaxy studies in crystal growth, [5] where it models the time evolution of the number of nuclei in non steady state crystal nucleation. Once the Φ -dependence is taken into account, the mutual information between the experimentally obtained distribution and the above second order corrected distribution is now 0.621 bits, that is, about 1 bit less than the previous estimate. Therefore, just 2.3% of the total information is not covered now by the second approximation.

4 Structure of the codebook and contrast distributions

The histogram of the σ distribution is depicted in figure 4. It has a drop around 100, due to the limited range of brightness levels, and also a cusp-like peak at 0, mostly due to many near flat blocks that arise from the layered structure of natural images. In fact, the 0 peak should be more marked if we were not discarding those patches B such that $\sigma_B \leq 4$. In any case, its structure is carried by the central linear zone, that suggests an exponential distribution. In fact, and as it could be expected, $p(\sigma)$ is quite correlated with the codebook

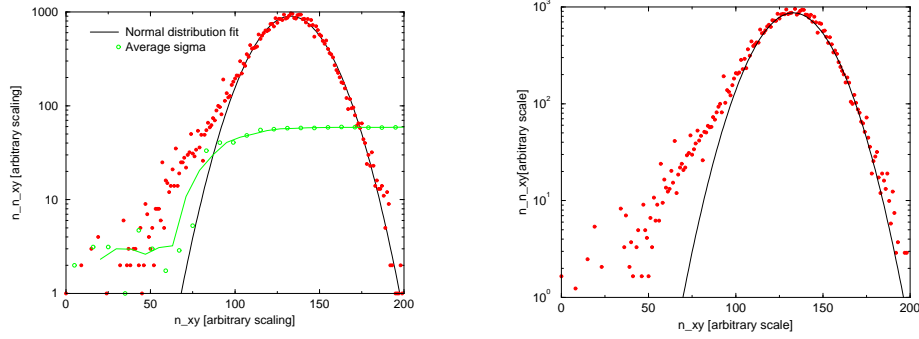


Fig. 5. Volume-corrected values of $\log N(m)$, with $N(m)$ the relative number of domains getting m natural blocks for the Lena (left) and van Hateren (right) codebooks. The distributions are fairly similar and suggest a volume-uniform distribution of all normalized natural patches among codebook domains. The left image also shows the average patch σ as a growing function of m . The scale of σ goes for a low value of about 5 at left, to a high value of at the right high count area.

source’s edges, as depicted in figure 4, left, that shows for the Lena image the values of $-\log p(\sigma(i, j))$ with $\sigma(i, j)$ the value for the domain with i, j as its right left corner coordinates.

To get a hold on the $p(i, j)$ distribution, we may count for each m' the number $N'(m')$ of domains that match exactly m' patches. In other words, we look at the (i, j) as a bins where matching patches fall. The resulting distribution has a clear parabolic structure, that suggests that domains “fall” more or less uniformly on bins or, more precisely, that in block space, patches are more or less distributed uniformly among domains. However, this uniform distribution hypothesis requires also to take into account the volume surrounding each domain when counting $N'(m')$, for then the probability of a codebook region \mathcal{R} receiving a patch is proportional to its volume $v(\mathcal{R})$. This would lead to the rather difficult problem of estimating this surrounding volume $v(D)$ for each domain. To avoid it, we have made the simplifying assumption that all hash linked lists correspond to domain space regions of the same volume and, therefore, to estimate $v(D)$ as a multiple of $\nu(h)$, the number of domains in the list indexed by $h = h(D)$. The a priori probability of a domain D is thus proportional to $\nu(h(D))$, and we should therefore correct the basic count m' of patches matched by a domain D to $m = m' \times \nu(h(D))$. The corrected $N(m)$ values are depicted in figure 5 in logarithmic scale, for the Lena (left) and van Hateren (right) codebooks. Both show a very similar parabolic structure, in which the left side divergence is due to low σ patches, with small denominators in (6) and, hence, more sensible to noise variations that may alter the matching domain they are assigned to. Notice that this effect should be more marked in domains getting fewer blocks. This is supported in figure 5, left, that shows for m the average σ value among the $N(m)$ patches: it is much smaller for small m .

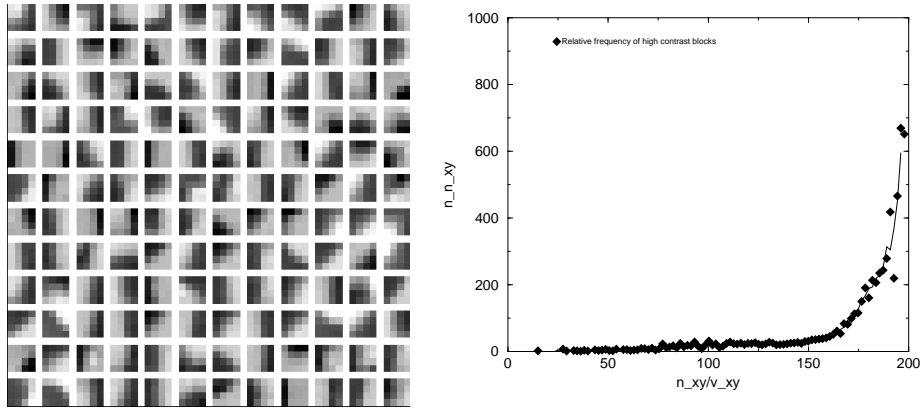


Fig. 6. Left: domains with a higher count of high contrast patches. They clearly correspond to edges. Right: the proportion of high contrast patches is markedly higher among high count domains.

It is clear that figure 5 is best explained as a large sample gaussian approximation of a binomial distribution or, in other words, that the codebook indices (i, j) do follow a volume-uniform distribution. Apparently this may contradict recent results in [4], that show a marked structure of high contrast natural 3×3 blocks. However, notice that figure 6, right, depicting the proportion of the high-contrast codebook domains getting a (normalized) number m of patches, has a very sharp rise at the high patch count area. Moreover, when the domains with the largest count of high contrast patches are depicted, as in figure 6, left, it is clear that they correspond to edges.

Finally, we just mention that the μ distribution is highly dependent on factors such as the camera’s calibration and can be easily manipulated through, say, histogram equalization. Thus, it does not carry significant information.

5 Summary and future directions

To alleviate the large dimension of the state space of 4×4 natural patches, we have proposed in this work to estimate their distributions in terms of an image compression inspired codebook approximation of the form $B \simeq (D_B, \sigma, \mu)$, with σ, μ the block’s variance and mean and D_B a codebook domain close to the normalization of B . Identifying a domain D_B in terms of its (i, j) location on the source image and the symmetry s applied, we have also shown how to factorize the distribution $p(i, j, s, \sigma, \mu)$ as $p(i, j, s, \sigma, \mu) \simeq p(i, j, s)p(\sigma)p(\mu)\Phi(\|\nabla B\|)$. Of these factors, the most relevant in terms of information seem to be $p(\sigma)\Phi(\|\nabla B\|)$ combination, which allows us to conclude that, at least in the scale investigated (about one minute of angle), the information is essentially carried by the block’s edges. This is certainly not surprising, as it agrees with the well known Marr

hypothesis [6]. However, this conclusion is achieved here through direct information theoretical considerations; in particular, they are independent of any consideration regarding the receiving system. In turn, this could suggest that biological systems have adapted themselves to extract those natural image parts most relevant in terms of information theory.

Moreover, the structure of the marginal distributions $p(i, j)$ and $p(\sigma)$ may have practical applications in areas such as image database searching. In fact, the absence of long tails in these distributions shows that the patches' representation proposed here has a very compact range. Thus, using the (i, j, σ) representation as a key for database searching, the worst distributed key is actually the exponentially distributed image contrast, while the search time for the other distributions should be nearly constant. Moreover, the very fast drop of the exponential distribution makes it reasonable to expect that the codebook coding scheme proposed here should allow for fast image database search strategies.

References

1. Y. Fisher (ed.), **Fractal Image Compression: Theory and Application**, Springer Verlag, New York, 1995.
2. J.H. van Hateren and A. van der Schaaf, *Independent component filters of natural images compared with simple cells in primary visual cortex*. Proc.R.Soc.Lond. B, 265:359-366, 1998.
3. K. Koroutchev and J. Dorronsoro, *Hash-like Fractal Image Compression with Linear Execution Time*, Iberian Conference on Pattern Recognition and Image Analysis, IbPRIA 2003, Lecture Notes in Computer Science 2652, 395-402, 2003.
4. A.B. Lee, K.S. Pedersen and D. Mumford. *The Complex Statistics of High-Contrast Patches in Natural Images*. In WWW Proceedings of Second International IEEE Workshop on Statistical and Computational Theories of Vision. Vancouver, Canada, July 2001.
5. I.M. Markov, **Crystal growth for beginners: Fundamentals of Nucleation, Crystal Growth and Epitaxy**, World Scientific, 1995.
6. D. Marr, **Vision**, W.H. Freeman and Co., 1982.
7. B.A. Olshausen, D.J. Field, *Natural image statistics and efficient coding*, Workshop on Information Theory and the Brain, Network, vol. 7.
8. L. Paninski, *Estimation of Entropy and Mutual Information*, Neural Computation, 15 (2003) 1191-1253.
9. K.S. Pedersen and A.B. Lee. *Toward a Full Probability Model of Edges in Natural Images*. In Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark. LNCS 2350, p. 328-, Springer-Verlag, May, 2002.
10. D.L. Ruderman, *The statistics of natural images*, Network: Computation in Neural Systems, vol. 5, 517-548, 1994.
11. A. Srivastava, A.B. Lee, E.P. Simoncelli and S.C. Zhu, *On Advances in Statistical Modeling of Natural Images*, Journal of Mathematical Imaging and Vision 18(1), 17-33, 2003.
12. M. Weinberger, G. Seroussi and G. Sapiro, *LOCO-I: A Low Complexity, Context-Based, Lossless Image Compression Algorithm*, Proc. of the IEEE Data Compression Conference, Snowbird, Utah, March-April 1996.