

**UNIVERSIDAD AUTONOMA DE MADRID**

**ESCUELA POLITECNICA SUPERIOR**



**Grado en Ingeniería de Tecnologías y Servicios de  
Telecomunicación**

**TRABAJO FIN DE GRADO**

**SISTEMA DE EXTRACCIÓN DE APUNTES EN PIZARRA  
CONVENCIONAL**

**Diego Díaz Rosado  
Tutor: Kostadin Koroutchev**

**JUNIO 2018**



# **SISTEMA DE EXTRACCION DE APUNTES EN PIZARRA CONVENCIONAL**

**AUTOR: Diego Diaz Rosado**  
**TUTOR: Kostadin Koroutchev**

**Escuela Politécnica Superior**  
**Universidad Autónoma de Madrid**  
**Junio de 2018**



# Resumen

Este trabajo de fin de grado tiene como objetivo principal diseñar y desarrollar un sistema para la grabación de tutorías con dispositivos convencionales de bajo coste, evitando así los dispositivos específicos y de alto coste utilizados por el método Didáctica Tutoría Universitaria con Soporte Hipermedia (DTUSH).

El sistema diseñado cuenta con una cámara web con micrófono integrado, que es posicionada delante de una pizarra convencional para realizar la grabación de la tutoría. El video generado por la cámara web es preprocesado llevando a cabo una estabilización que compensará el movimiento que se haya podido producir en la cámara junto con un filtrado, que adecua la secuencia de video para ser procesado por el algoritmo encargado de la eliminación del primer plano en la secuencia de video.

El algoritmo desarrollado trabaja sobre los frames del video a color, analizando los cambios producidos entre frames consecutivos para realizar la eliminación del primer plano creando una secuencia de video con la escritura realizada en la pizarra.

La secuencia creada por el algoritmo en esta fase de procesamiento no contendrá el audio de las explicaciones dadas durante la tutoría, por lo que se llevará a cabo un proceso de posprocesado que permite añadir el audio del video original. En esta última fase del proceso se extraerá el audio del video original y se incrustará en el video sintético creado, de tal manera, que las explicaciones dadas por el tutor durante la tutoría, junto con los trazos de la escritura en la pizarra, se muestren de manera síncrona en la secuencia de video.

El resultado de este proceso generará un material hipermedia reutilizable, con un formato de video abierto, en disposición de ser reproducido por cualquier reproductor multimedia o ser cargado en una plataforma de video online para navegar a través de su contenido.

## Palabras clave

Tutoría, pizarra, detección de movimiento, compensación de movimiento, indexación, procesado de vídeo, análisis secuencias de video, massive online open courses (MOOCS)

# Abstract

This Bachelor Thesis main objective is to design and develop a system for recording meetings with conventional and low-cost devices, avoiding specific devices with a high cost used by DTUSH method.

The designed system has a webcam with a microphone which is located in front of a whiteboard in order to record meetings. The video produced by the webcam is processed carrying through a stabilization and will make up the movement that could have been produced in the camera with a filtrate which adopts the video sequence to be processed by the algorithm responsive for the elimination of the foreground in the video sequence.

The developed algorithm works on color video frames, analyzing produced changes between consecutive frames in order to delete foreground through a shift, which means creating video sequence with the script on the whiteboard.

The sequence which has been created by the algorithm during this process period will not include reason audios during the meeting, carrying a post-processed process which allow adding the audio of the original video. In the last stage of the process removing the audio from the original video and embedding this into a synthetic video which will be created, as the reasons which are given by the tutor during the meeting as script strokes the whiteboard, will show simultaneously during the video sequence.

The result of this process will produce a hypermedia resource material with an open audio format, that it could be reproduced by any multimedia device or to be uploaded in an online video platform to be able to search through its contents.

## Keywords

Meeting, whiteboard, tracking movement, index, video processing, video analitics, massive online open courses (MOOCS)

## ***Agradecimientos***

*Me gustaría dar las gracias en primer lugar a Kostadin, por haberme permitido trabajar con él durante todo este tiempo y por toda la ayuda que me ha prestado y lo mucho que he aprendido.*

*También agradecer a mi familia, en especial a mis padres, hermana y abuelos, haberme apoyado durante todo este tiempo en los malos y buenos momentos como siempre han hecho.*

*Y como no, estar muy contento de haber conocido a personas que me han acompañado todos estos años y que hoy son amigos: Guillermo, Peli, Iván, Ricardo, Guille Luna, Mario, Raúl, Riki, Gonzalo.*

*A todos,*

*Gracias.*





# INDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Motivación.....	1
1.2	Objetivos.....	1
1.3	Organización de la memoria.....	2
2	Estado del arte .....	3
2.1	Método de tutoría con soporte hipermedia DTUSH.....	3
2.1.1	Dispositivos. ....	3
2.1.2	Metodología.....	4
2.1.3	Evaluación. ....	5
2.1.4	Conclusiones.....	5
3	Diseño.....	7
3.1	Introducción.....	7
3.2	Arquitectura del sistema .....	7
3.2.1	Adquisición de video .....	8
3.2.2	Preprocesado de video .....	9
3.2.3	Procesado de video .....	10
3.2.4	Posprocesado de video .....	10
4	Desarrollo .....	11
4.1	Algoritmo desarrollado.....	11
5	Pruebas y resultados .....	21
6	Conclusiones y trabajo futuro.....	23
6.1	Conclusiones.....	23
6.2	Trabajo futuro .....	23
	Referencias .....	25
	Glosario .....	27
	Anexos.....	- 2 -
A	Manual de instalación.....	- 2 -
B	Manual del programador .....	- 4 -

## INDICE DE FIGURAS

FIGURA 2-1: LIVESCRIBE ECHO.....	3
FIGURA 2-2: DOT PAPER LIVESCRIBE.....	3
FIGURA 2-3: EJEMPLO MATERIAL HIPERMEDIA TUTORIA.....	4
FIGURA 2-4: LECTURA HIPERTEXTUAL Y ESTRUCTURA SOPORTE HIPERMEDIA.....	5
FIGURA 3-1: ARQUITECTURA DEL SISTEMA.....	6
FIGURA 3-2: CÁMARA WEB GENIUS WIDECAM F100 .....	7
FIGURA 3-3: MODELO PROCESAMIENTO DE VIDEO.....	8
FIGURA 3-4: ARQUITECTURA DESPLEGADA DEL SISTEMA.....	9
FIGURA 4-1: ORGANIGRAMA ALGORITMO.....	10
FIGURA 4-2: MARCADORES DELIMITANTES DE PIZARRA.....	11
FIGURA 4-3: FUNCIONAMIENTO FILTRADO MEDIANA.....	12
FIGURA 4-4: ACELERACIÓN FILTRADO MEDIANA.....	12
FIGURA 4-5: FRAME ORIGINAL.....	13
FIGURA 4-6: FILTRADO MEDIANA CON VECINDARIO DE TAMAÑO 15.....	13
FIGURA 4-7: FILTRADO GAUSSIANO SIGMA 2.....	14
FIGURA 4-8: RESTA FILTRADO MEDIANA Y GAUSSIANO.....	14
FIGURA 4-9: SEGUNDO PROCESO FILTRADO.....	15
FIGURA 4-10: EVOLUCION VALORES PIXEL.....	15
FIGURA 4-11: EVOLUCIÓN VALORES PIXEL ESCRITURA.....	16
FIGURA 4-12: EVOLUCION VALORES PIXEL MANO.....	16
FIGURA 4-13: MEDIA DESPLAZANTE.....	17

# 1 Introducción

---

## 1.1 Motivación

La tutoría universitaria es uno de los procesos docentes que juegan un papel importante en el aprendizaje de los alumnos. Es un método de carácter individual o dirigido a grupos reducidos, en los que la transferencia de conocimientos se adapta a los problemas o dudas de los estudiantes con dinámicas de conversación. Podríamos decir que puede ser considerado como uno de los métodos más eficientes en sentido de entendimiento y transferencia de conocimiento.

Las técnicas tradicionales de realizar tutorías realizando explicaciones y resolviendo dudas con anotaciones en papel, o en pizarras convencionales, son técnicas orientadas uno a uno y sin posibilidad de que el conocimiento transmitido individualmente sea compartido o que las explicaciones dadas sean reutilizables.

Fijándose en estos problemas, M. Francisco Aguilar Tamayo y Nehemías Moreno Martínez diseñan el sistema Didáctica Tutoría Universitaria con Soporte Hipermédia (DTUSH) [1], para poder almacenar todas las explicaciones tanto las escritas como las de audio expuestas en la tutoría el método utiliza un lápiz óptico y cuaderno de papel digital, que permiten la grabación de todas las explicaciones y detalles que da la tutoría al estudiante. Los datos recogidos durante la sesión de tutoría forman un material hipermédia que puede ser consultado transfiriéndolo a un ordenador, tablet o smartphone, de esta manera el contenido de la tutoría se convierte en reutilizable.

El principal problema que plantea el método DTUSH son los altos costes que suponen la adquisición de los dispositivos y materiales especializados.

## 1.2 Objetivos

El objetivo de este trabajo es diseñar un sistema que evite los dispositivos especializados que emplea la metodología DTUSH.

Para ello, únicamente es necesario una cámara web posicionada delante de la pizarra convencional. Este diseño es adaptable a cualquier sala o despacho del profesorado permitiendo un uso generalizado del DTUSH hardware estándar de bajo coste.

Se desarrollará un algoritmo para procesar el video capturado trabajando sobre las secuencias de video a color y con una tecnología que consistirá en:

1. Grabación de video con audio sincronizado.
2. Los videos grabados se procesan eliminando el Foreground (FG) es decir, las personas, la mano con la que el profesor escribe, el rotulador utilizado, etc.
3. El resultado será transformado a un formato abierto, ofreciendo la posibilidad de extender el método con otro tipo de dispositivos y entornos de programación para el desarrollo de métodos analíticos sobre el contenido de las tutorías.

### **1.3 Organización de la memoria**

La memoria consta de los siguientes capítulos:

- [Capítulo 1](#). Introducción: Motivación, objetivos y organización de la memoria.
- [Capítulo 2](#). Estado del arte: Método DTUSH, recursos hipermedia, metodología, evaluación, conclusiones.
- [Capítulo 3](#). Diseño: Introducción, arquitectura del sistema, adquisición de video, preprocesado, procesado, postprocesado.
- [Capítulo 4](#). Desarrollo: Algoritmo desarrollado.
- [Capítulo 5](#). Pruebas y resultados.
- [Capítulo 6](#). Conclusiones y trabajo futuro.
- [Bibliografía](#).

## 2 Estado del arte

---

### 2.1 Método de tutoría con soporte hipermedia DTUSH

El método DTUSH, consiste en la utilización de un lápiz electrónico y un cuaderno digital, que permiten grabar la escritura y el audio de una tutoría, generando un recurso hipermedia que es accesible tanto para estudiantes, como para tutores. Proponiendo así, una didáctica centrada en el dialogo y la conversación académica para el desarrollo de tutorías con estudiantes como se describe a continuación en la metodología (capítulo 2.1.2)

#### 2.1.1 Dispositivos.

El hardware utilizado para obtener la escritura y el audio, es un lápiz electrónico de la marca Livescribe. Este dispositivo debe ser utilizado con un papel punteado específico que contiene puntos de referencia que permiten al lápiz electrónico calcular y almacenar la secuencia temporal de las posiciones de la hoja donde los trazos son escritos.

Esta escritura, junto con el audio capturado a través del micrófono, forma el contenido hipermedia que es almacenado en el dispositivo para su posterior transferencia al ordenador, tablet, o smartphone, y que permitirá una consulta en pantalla por medio del software propietario de la marca.

La visualización es mostrada como un video de los trazos dibujados, sincronizados con el audio capturado. También existe la posibilidad de exportar las acciones escritas del contenido hipermedia como un archivo PDF. [1]

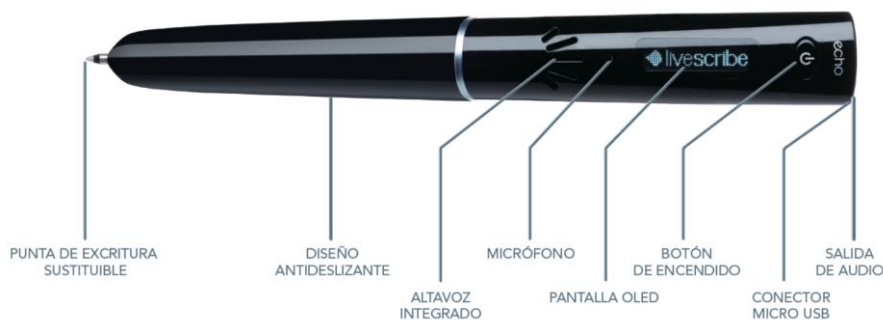


Figura 2-1 Livescribe Echo.



Figura 2-2 Dot Paper Livescribe

## 2.1.2 Metodología.

Las sesiones de tutoría podían desarrollar unas dinámicas distintas dependiendo del tutor, pero en ellas habitualmente se podían distinguir tres fases:

### 1. Inicio de la sesión:

El tutor pide al estudiante exponer las razones por las que ha pedido una tutoría, estas pueden exponerse mediante preguntas o planteamiento de problemas. Durante este proceso, la función del tutor es comprender el problema y tomar notas para preparar un guion sobre el desarrollo de la tutoría, así como detectar problemas y dificultades que serán atendidas durante la sesión. Ello permitirá conocer las necesidades del estudiante. Otro aspecto a considerar es que estas notas tomadas al inicio de la sesión serán consultadas posteriormente por el estudiante y podrán servir como organizadores para la consulta de la tutoría, que no será una simple grabación sino un material hipermedia.

### 2. Desarrollo de la sesión:

La tutoría se desarrolla en forma de dialogo, la dinámica de la conversación no es el mismo que el de una clase en grupo o una conferencia, está regulado según los procesos de entendimiento entre ambos participantes. Además de las palabras, se acompaña del discurso oral, notas, esquemas, gráficos y texto que permiten una consulta específica para recuperar información.

### 3. Cierre de la sesión:

Una vez discutidos los temas de la tutoría, se incluyen las tareas y compromisos, que permiten fijar las metas para el proceso de aprendizaje para la siguiente sesión.

El cierre de la sesión permite recapitular la discusión desarrollada y ofrecer ideas para la consulta o lectura de la tutoría hipermedia.

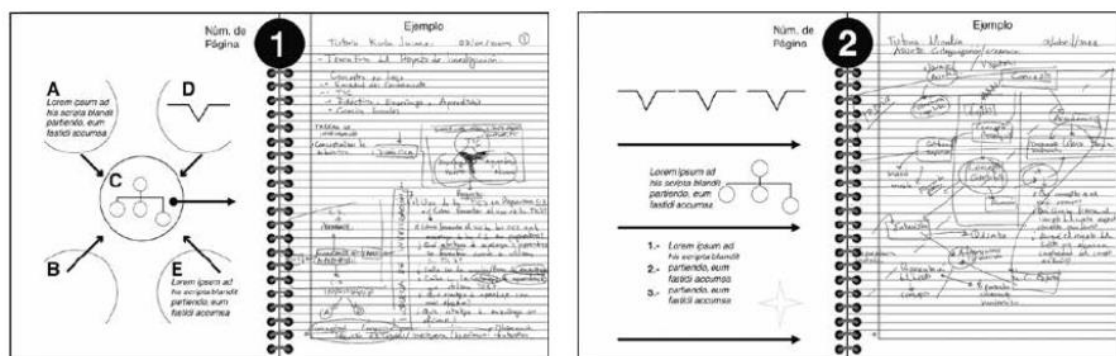


Figura 2-3 Ejemplo material hipermedia de tutoría

### 2.1.3 Evaluación.

Una vez terminada la tutoría y el proceso de grabación, tanto tutor como estudiante tienen acceso al registro de la tutoría, con recursos como imagen final, imagen dinámica y grabación de voz, para así, acompañar el aprendizaje y la reflexión de su proceso formativo.

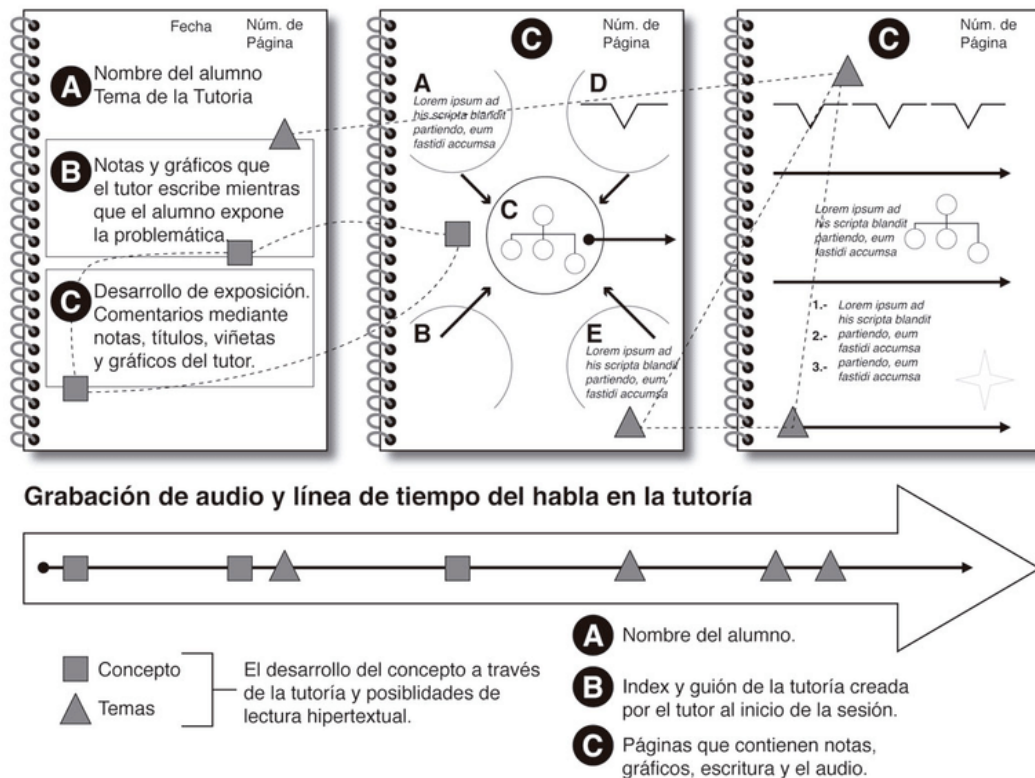


Figura 2-4 Lectura hipertextual y estructura de la tutoría con soporte hipermedia.

### 2.1.4 Conclusiones

Este método parece mejor a otros por aspectos destacables, como que la tutoría deja de tener un carácter puntual, e irrepitable, pasando a ser reproducible y más colaborativo y participativo [1]. El estudiante y tutor tendrán la posibilidad de elegir un segmento del video y escuchar las explicaciones específicas para esta parte de la tutoría, tomar notas durante la reproducción e incluso realizar una impresión de estas para una posterior consulta.

También si se trata de una asignatura no individualizada, otros alumnos o estudiantes pueden tener dudas o problemas relacionados con la tutoría grabada y tendrán la posibilidad de acceder al recurso hipermedia y visualizarlo para así conseguir resolverlas, este carácter reutilizable, ofrece la posibilidad de estar completamente ausente de la misma, siendo de utilidad para la educación a distancia y los contenidos online como cursos masivos (MOOCS).

El método DTUSH plantea la desventaja del uso de dispositivos privativos de una marca.





# 3 Diseño

---

## 3.1 Introducción

En esta parte se describe el diseño en que se ha basado el sistema creado para satisfacer los objetivos propuestos, utilizando únicamente una cámara web como dispositivo para la adquisición de las imágenes para cada instante temporal de una secuencia de video en una tutoría cualquiera que se realice en una sala que cuente con una pizarra convencional.

## 3.2 Arquitectura del sistema

Se procede a seguir con el diseño que tendrá tal sistema para cumplir con los objetivos buscados. En este apartado se definirán las funcionalidades de cada módulo del sistema y el fin con el que han sido creados, sin profundizar en el funcionamiento específico que se verá de manera detallada en el apartado de desarrollo (capítulo 4)

Se ha considerado un sistema compuesto por tres bloques principales desde que se dispone del video hasta que se crea el video final con la eliminación del primer plano en sincronía con el video origen.

El primero de ellos se encarga del preprocesado de video, donde se abarcan las tareas de estabilización de imagen y filtrado de la secuencia. Estos tratamientos son realizados sobre la secuencia original dando lugar a un video estable y filtrado que será procesado para llevar a cabo la extracción del primer plano.

El video sin el primer plano no dispondrá de audio, que será extraído del video original para ser añadido al video sintético de manera síncrona en el bloque de posprocesado, dando lugar al video final, sin el primer plano y con audio.

A continuación, se puede observar la arquitectura que sigue este proceso en el esquema de la siguiente figura:

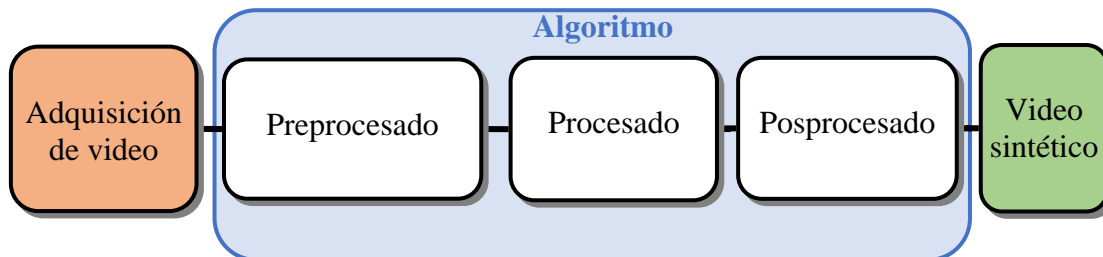


Figura 3-1: Arquitectura del sistema

### 3.2.1 Adquisición de video

Para la adquisición de las secuencias de video, el sistema cuenta con una cámara web de la marca Genius, posicionada en la pared, delante de la pizarra convencional, correctamente enfocada a la distancia de la pizarra y con un ángulo de visión que permita para abarcar todo el alto y ancho de la pizarra y por tanto los marcadores delimitantes para realizar la grabación de video. Este dispositivo de grabación cuenta con un micrófono estéreo para la captura del audio durante toda la sesión

Las especificaciones se detallan a continuación:

- Fabricante KYE Systems America Corporation.
- Modelo WideCam F100
- Sensor de imagen CMOS de píxeles Full HD 1080p
- Tipo de lente objetivo de enfoque manual
- Ángulo de visión horizontal 120°
- Formato archivo de vídeo MPEG/WMV.
- Grabación a color.
- Resoluciones (DPI) 12MP (interpolación), 1920 x 1080, 1280 x 720, 640 x 480 píxeles.
- Peso 82 g.
- Dimensiones (A x A x P) 150 x 49 x 48 mm
- Micrófono estéreo integrado.
- Conectividad a través de USB 2.0.
- Compatibilidad Mac, PC.



**Figura 3-2: Cámara Web Genius WideCam F100**

### 3.2.2 Preprocesado de video

El módulo de preprocesamiento de video cuenta con un estabilizador de imagen a nivel de pixel, que trabaja a partir de unos marcadores que dibujados en la pizarra antes del comienzo de la sesión delimitarán las esquinas superiores e inferiores de la pizarra. Este estabilizador de imagen, compensa el movimiento que se puede producir durante la tutoría por razones ambientales.

La grabación puede ser realizada con cualquier dispositivo de grabación, siempre y cuando se tenga en cuenta que en el ángulo de visión de la cámara deben aparecer estos marcadores.

Una vez estabilizada la secuencia de video se lleva a cabo el proceso de filtrado sobre todos los frames de video para eliminar la información no deseada producida en el proceso de adquisición.

Se busca una eliminación del ruido gaussiano generado por el sensor de la cámara, o variaciones de iluminación y/o altas temperaturas, ya que la intensidad de todos los pixeles se ve afectada, debido a esto, se opta por un filtrado gaussiano, que produce un suavizado más uniforme que el filtro de media produciendo un “emborronado” como se puede apreciar en la imagen de la figura 4-5.

Al igual que se realiza un filtrado gaussiano, a los mismos frames también se les realiza un filtrado de mediana ya que la meta es la reducción de ruido y la preservación de los bordes [2].

Este procesamiento se puede visualizar en la figura 4-4.

A estos dos filtrados realizados en cada frame se les aplicará la resta entre ambos para lograr la mejora significativa en la reducción de ruido.

Este mismo proceso es realizado una segunda vez, como se define en la figura 3-3 para optimizar aún más los resultados de cara al procesado para el proceso de extracción de del primer plano.

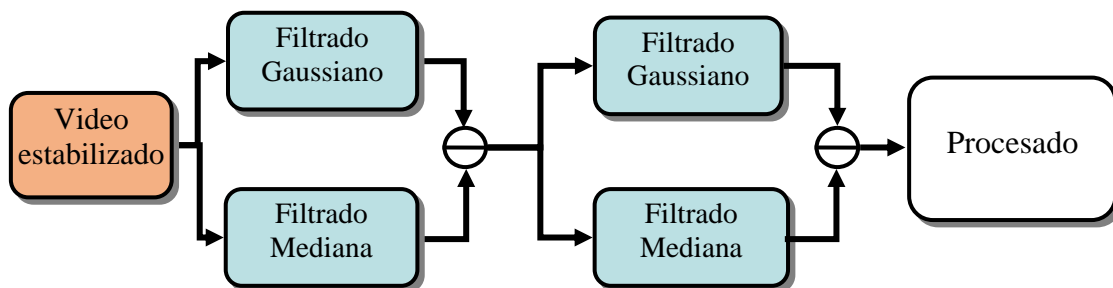


Figura 3-3: Módulo procesamiento de vídeo

### 3.2.3 Procesado de video

Este módulo realiza la eliminación del primer plano en la secuencia de video a color que ha sido previamente estabilizada y filtrada, el funcionamiento del algoritmo para realizar dicha tarea será descrito en profundidad en el apartado cuatro del desarrollo.

### 3.2.4 Posprocesado de video

Lleva a cabo el añadido de audio al video obtenido a partir de la eliminación del primer plano en todo los frames que lo componen.

Para la inserción del audio será necesario realizar una extracción del audio perteneciente al video original, atendiendo a la frecuencia de muestreo con la que fue grabada la señal de audio contenido en él vídeo. En el proceso de escritura para la creación del video sintético final la señal de audio extraída será añadida en correspondencia con de la tasa de frames con la que contará el video sintético final para obtener una sincronía entre ambos.

El video sintético creado se graba en formato AVI que permite el posterior uso en la mayoría de los sistemas actuales para posibles modificaciones sobre el mismo.

El detalle del proceso seguido para el diseño se puede observar a continuación en la figura 3-4

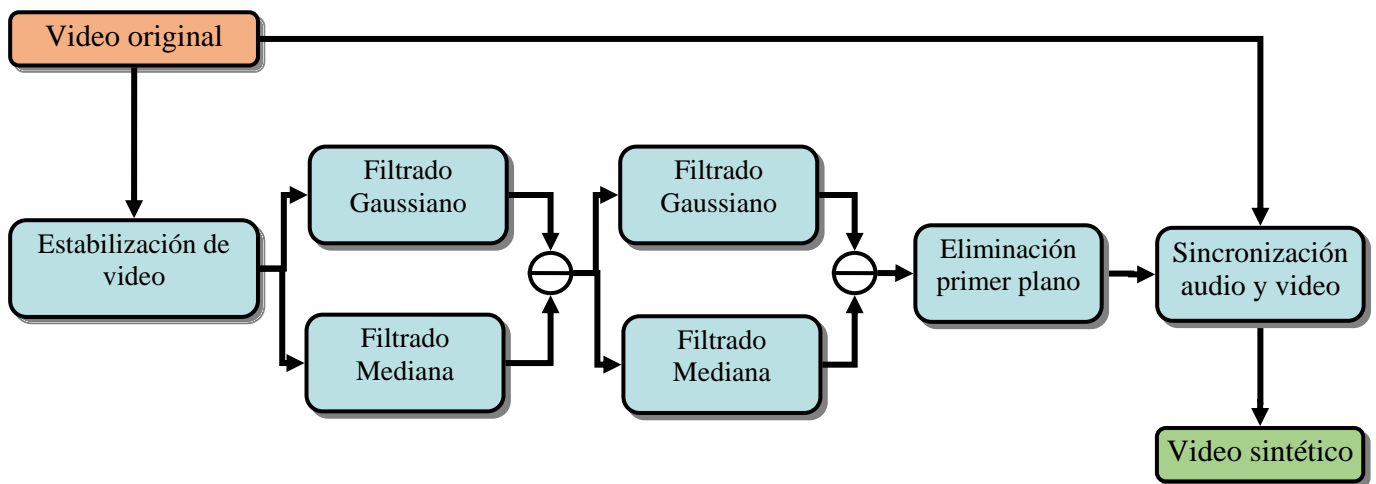


Figura 3-4: Arquitectura desplegada del sistema

# 4 Desarrollo

---

## 4.1 Algoritmo desarrollado

El algoritmo se ha implementado íntegramente en Matlab, desarrollando un conjunto de funciones para cada módulo que compone el sistema diseñado, desde el preprocesamiento del video, pasando por el procesado hasta el módulo de sincronización. Esto ofrece la posibilidad de poder integrar cualquier módulo nuevo o de mejora de uno ya existente sin necesidad de modificar el sistema al completo.

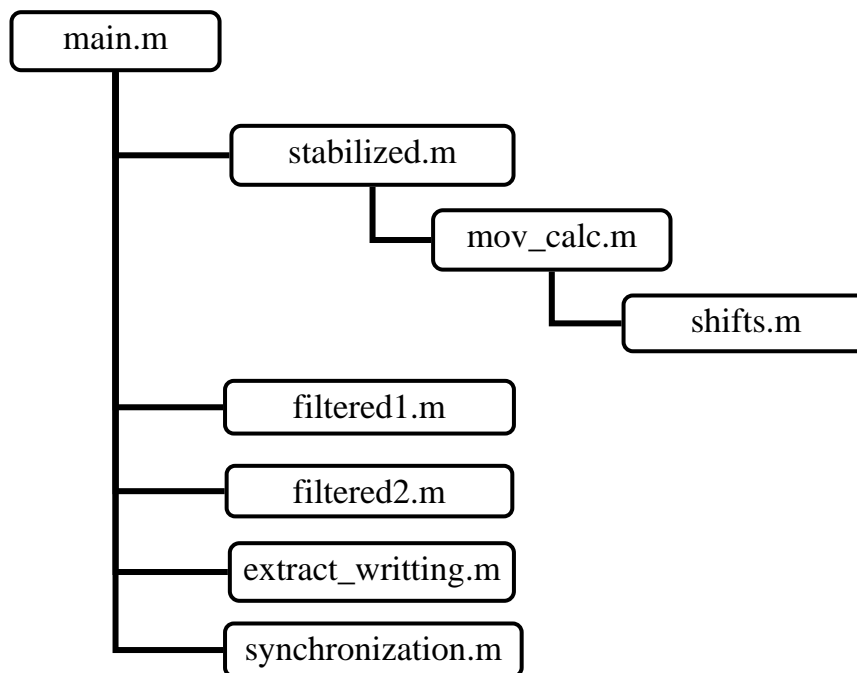


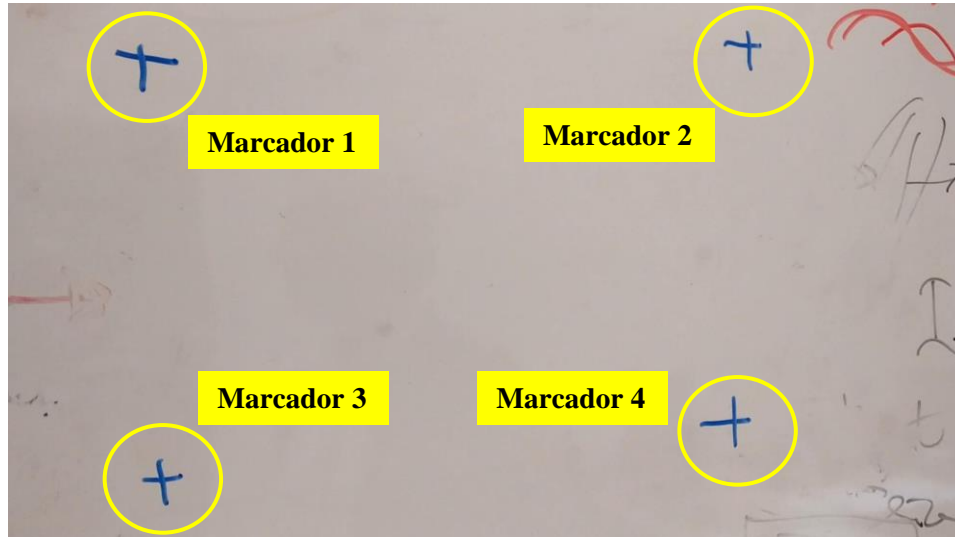
Figura 4-1: Organigrama algoritmo

En la función principal se indican los procesos que se realizan sobre el video obtenido desde la cámara web, siguiendo el orden de ejecución de la figura 3-4 del capítulo anterior.

Una vez capturado el video este deberá ser colocado en el directorio raíz donde se encuentre el archivo **main.m** para una correcta ejecución.

El proceso de estabilización es el primero en ejecutarse. El algoritmo de estabilización se lleva a cabo utilizando el principio de correlación por máscaras, utilizado frecuentemente para medir similitudes entre imágenes o partes de ellas, como ocurre en el reconocimiento de patrones.

La función **stabilized.m** se encarga de realizar la estabilización del video a nivel de pixel, compensa el movimiento que se haya podido producir durante la grabación, para ello son definidos unos marcadores delimitantes como se muestra en la figura 4-2.



**Figura 4-2: Marcadores delimitantes de pizarra.**

Los marcadores delimitantes de la pizarra en el caso de la figura son unas cruces, pero podrían ser cualquier marca u objeto, modificando en el algoritmo se define cual es el área de búsqueda. En la versión original las coordenadas de los marcadores son introducidas a mano.

Estos marcadores son analizados en todos los frames que componen el video por la función **mov\_calc.m** encargada de almacenar los desplazamientos inter-frames de los marcadores, la función recibe un vector con las posiciones iniciales de la situación de los marcadores en el primer frame para que la función **shifts.m** calcule que desplazamientos se han producido desde la posición inicial del marcador, respecto de los frames siguientes, esto se realiza desplazando el marcador de referencia a nivel de pixel hasta conseguir que la correlación sea máxima, este desplazamiento será devuelto a la función **mov\_calc.m** para almacenarlo en un cell, y que la función **stabilized.m** realice el desplazamiento pertinente a nivel de pixel por medio de una transformación geométrica, para ello, se utilizarán los puntos de referencia y los puntos de desplazamiento calculados para cada frame de la secuencia que dará como resultado una compensación del movimiento sufrido por la cámara.

Como cabe la posibilidad de que se oculten alguno de los marcadores al realizar la tutoría, ya sea por que pase alguna persona por delante o porque la misma mano de la persona que escribe los pueda ocultar, se ha definido que serán necesarios que como mínimo dos de estos cuatro marcadores tienen que ser visibles para un correcto proceso de estabilización, por esta razón es razonable de situarlos en la parte superior y esquinas de la pizarra.

Una vez estabilizado el video, se ha visto que para que el algoritmo de eliminación del primer plano funcione correctamente, es necesario eliminar las imperfecciones de manera selectiva y de resaltar el contenido escrito, para esto, se ha optado por la utilización de técnicas de filtrado paso bajo que serán desarrolladas en la función **filtered1.m**.

Estas técnicas de filtrado consisten en la aplicación de una matriz de filtrado  $N \times N$  a cada uno de los pixeles de la imagen para generar un nuevo valor mediante una función aplicada al valor original del pixel y sus vecinos. Siendo el resultado final dividido entre un escalar.

Se ha optado por el uso de un filtro de mediana [2] para mantener el valor final del pixel con un valor real presente en la imagen, además de ser un filtrado que es menos sensible a valores extremos, esto emborronará la imagen para coger el color de la pizarra, de la manera mostrada en el ejemplo de la figura 4-3.



**Figura 4-3: Funcionamiento filtrado mediana**

Este filtrado es realizado sobre todos los frames del video original estabilizado, con un tamaño de vecindario 60, pero como se realiza sobre una imagen reducida en 1/3 el argumento de entrada es de 20, que ordena los valores en la vecindad de cada punto de menor a mayor, obteniendo el valor de la posición intermedia.

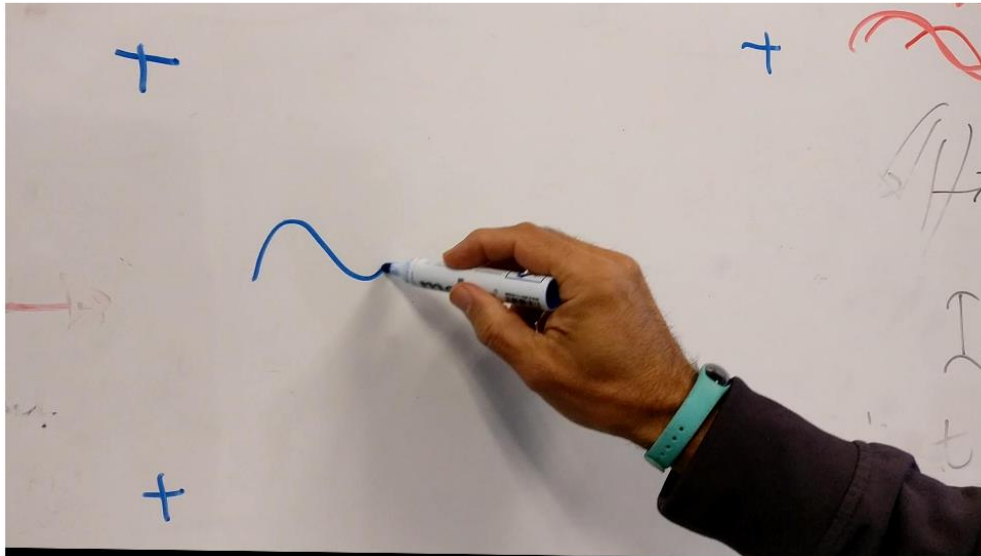
La dimensión de con la que cuenta cada frame es de 1920 x 1080 pixeles lo que implica que la realización del cálculo para la mediana lleva a cabo tiempos de computación muy altos. Esta razón lleva a buscar una solución de aceleración, cuya implementación consiste en reducir cada frame a un tercio respecto del original y realizar el cálculo de la mediana por cada componente de color en unas dimensiones de 360 x 640 pixeles. El proceso de aceleración se puede ver implementado en las siguientes líneas:

```
% The image is reduced to speed up the calculation of medians
tf3=imresize(thisFrame,1/3);
if(i==1)
    ilmed3=tf3;
end
% Median filtering for each component
med_size=15;% Neighborhood size for median filtering
ilmed3(:,:,1)=medfilt2(tf3(:,:,1),[med_size med_size]);
ilmed3(:,:,2)=medfilt2(tf3(:,:,2),[med_size med_size]);
ilmed3(:,:,3)=medfilt2(tf3(:,:,3),[med_size med_size]);
%Back to original size
ilmed=imresize(ilmed3,3);
```

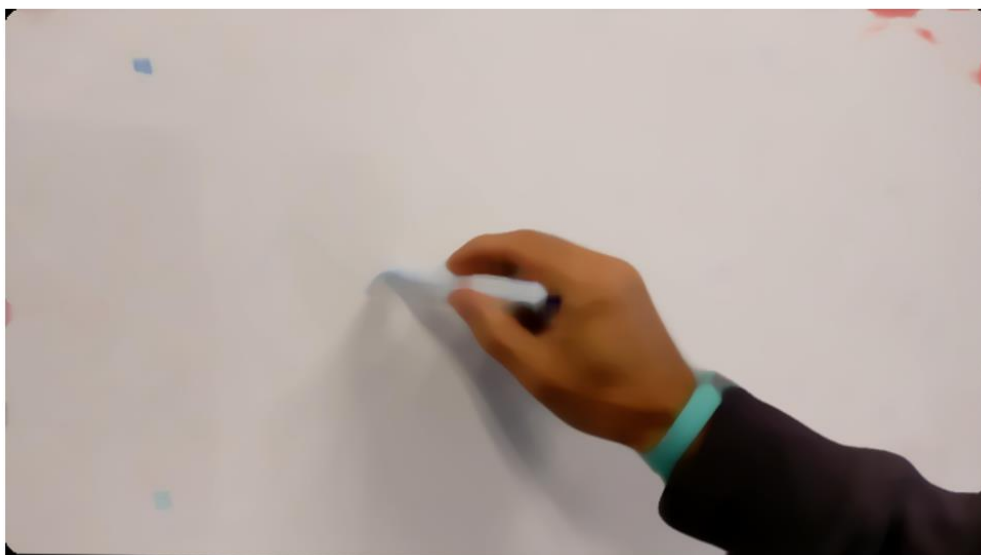
**Figura 4-4 Aceleración filtrado mediana**

Una vez realizado el cálculo por cada componente se devuelve el frame a su tamaño original de 1920x1080, para no modificar las dimensiones cuando se realicen operaciones sobre los mismos.

El resultado tras aplicar el filtrado de mediana al frame original del que disponíamos se puede observar en la figura 4-5 y la figura 4-6.



**Figura 4-5: Frame Original**



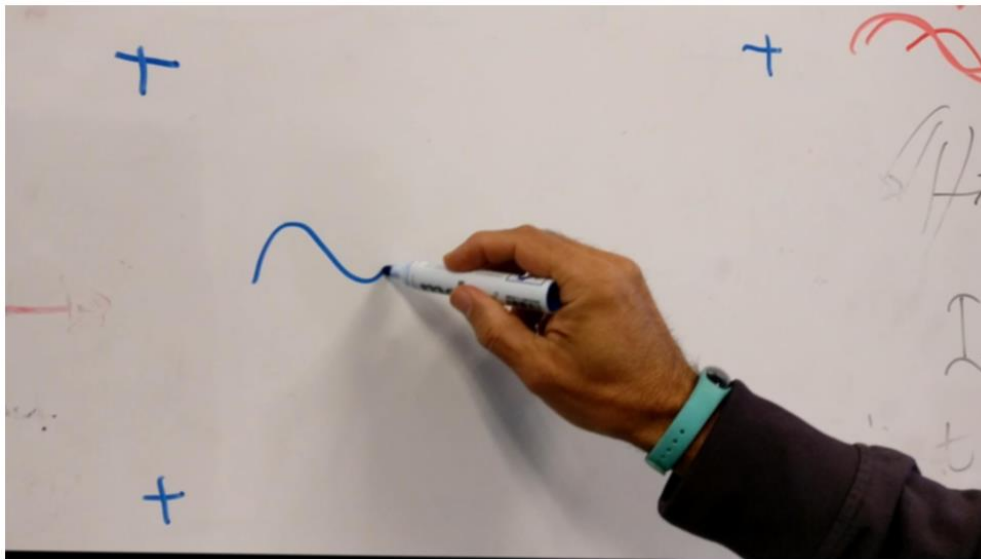
**Figura 4-6: Filtrado mediana con vecindario de tamaño 60.**



Paralelamente al filtrado en mediana, es aplicado un filtrado lineal gaussiano que sigue una distribución con la siguiente forma:

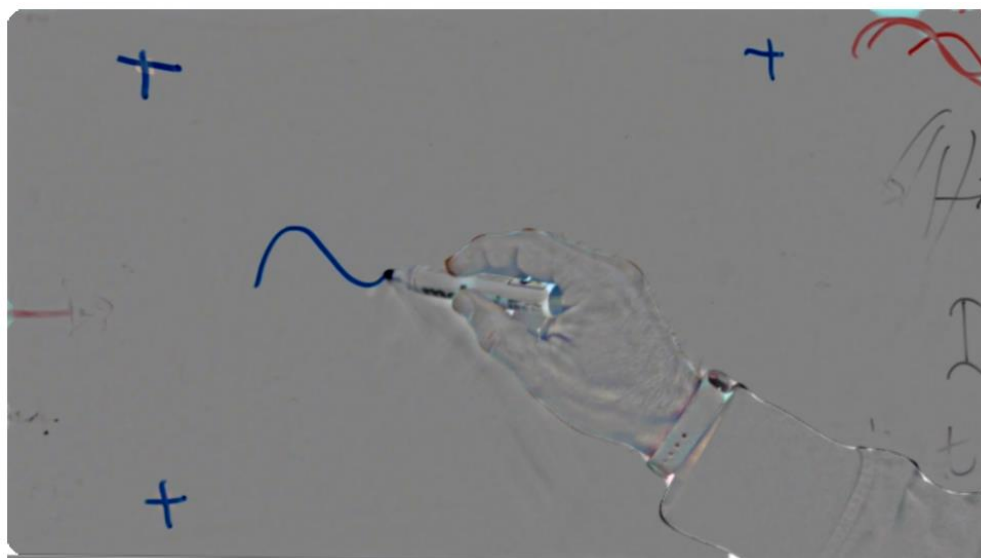
$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Para producir un suavizado más uniforme disminuyendo la nitidez, y aumentando la borrosidad, se ha realizado el filtrado gaussiano con una desviación estándar de  $\sigma=2$  para aportar un suavizado uniforme:



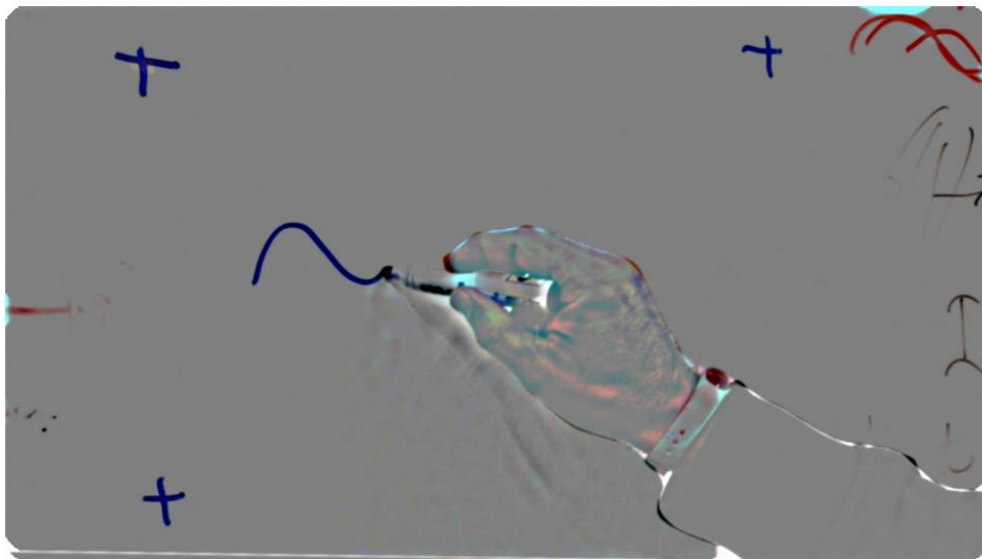
**Figura 4-7: Filtrado Gaussiano  $\sigma=2$**

Una vez obtenidos los dos filtrados de mediana y gaussiano se realizará una resta de los mismos para obtener un resultado como el de la figura 4-8



**Figura 4-8: Resta filtrado mediana y gaussiano.**

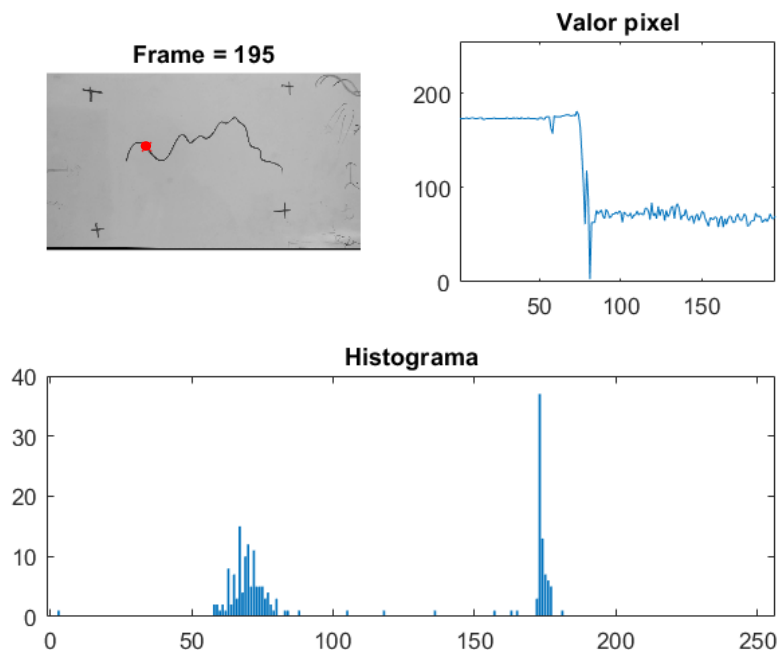
Para mejorar los resultados que se habían obtenido de este primer filtrado de media y gaussiano se ha optado por repetir el filtrado gaussiano para obtener una mejor aproximación a los resultados deseados que se muestran en la figura 4-9.



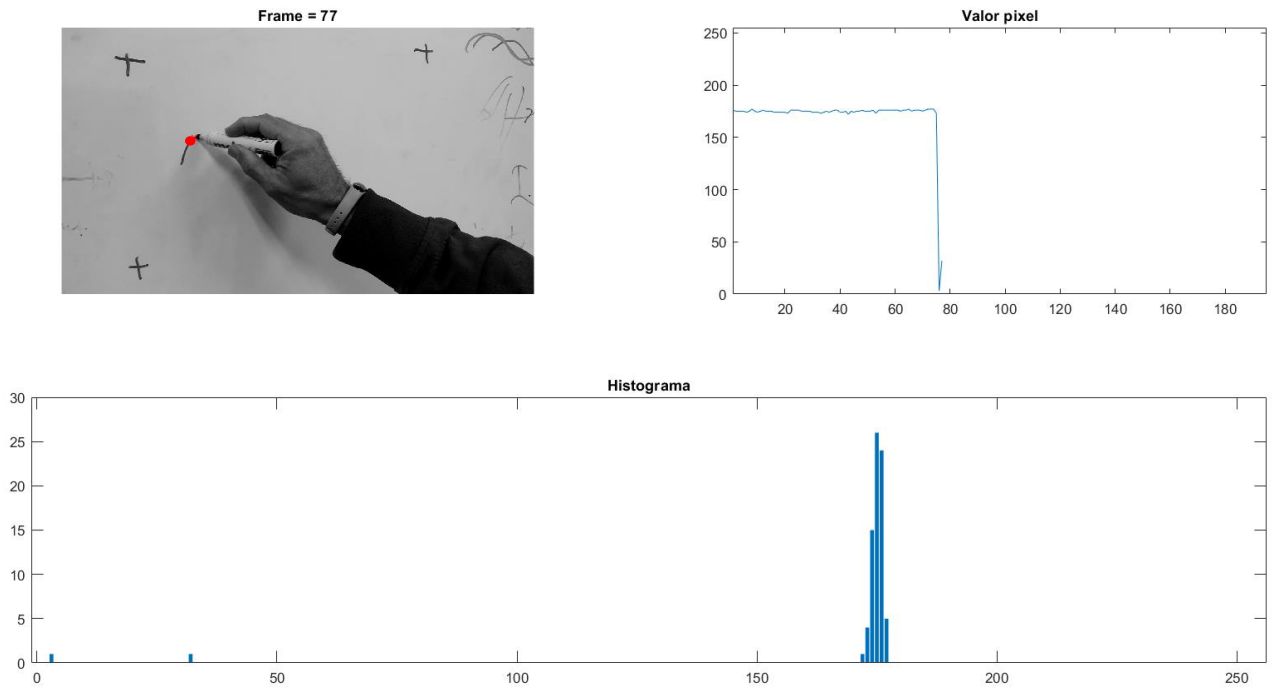
**Figura 4-9: Segundo proceso de filtrado**

Para la eliminación del primer plano en los frames de la secuencia de video, se ha realizado un estudio para observar cómo se producen los cambios a nivel de pixel cuando se escribe sobre la pizarra a lo largo de la secuencia de video.

Para ello se ha prestado atención al valor que toman los píxeles durante la transición de un frame a otro, desde el valor que posee un pixel cuando la pizarra está blanca completamente, hasta que se produce escritura en ella por medio de un rotulador. En la figura 4-10 se puede observar la evolución que sigue un pixel en escala de grises.

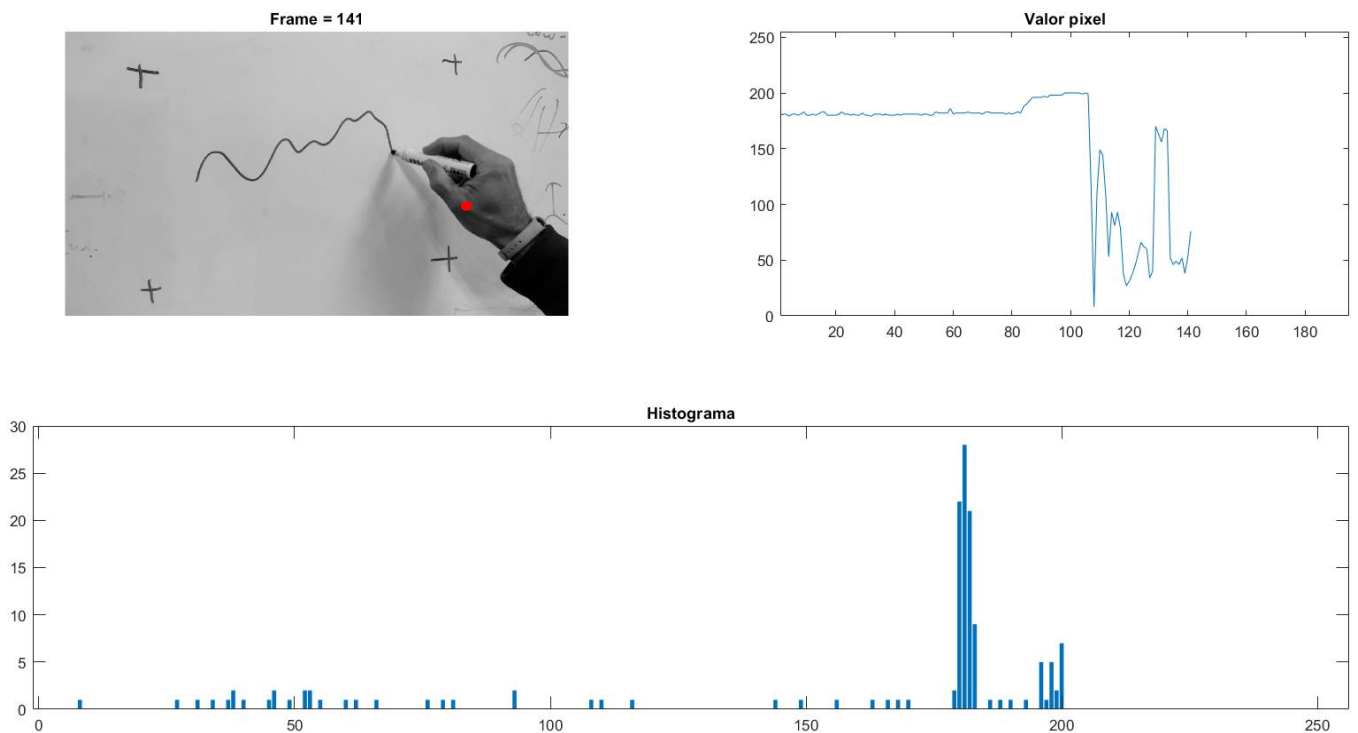


**Figura 4-10: Evolución valores pixel**



**Figura 4-11: Evolución valores pixel escritura**

El valor representante de la pizarra blanca es el que se sitúa en torno a 173 y el valor de la escritura en pizarra el que está en torno a 70. Por tanto, todos los valores pertenecientes al intervalo entre 70 y 173 no formarían parte de la región de interés que se busca preservar, siendo estos valores de pixel pertenecientes a la mano que está escribiendo, la persona o cualquier otro objeto que no tenga los valores de pizarra o escritura.



**Figura 4-12: Evolución valores pixel mano**

Tras los anteriores datos se puede concluir que lo que buscamos es eliminar la transición desde el valor de la pizarra blanca al de la pizarra escrita, buscando cual sería el valor de cambio consiguiendo una buena aproximación mediante de la siguiente relación:

$$V_s = \frac{V_1 + V_\infty}{3}$$

Siendo  $V_1$  el valor que toma el pixel al comienzo, y  $V_\infty$  el valor que toma el pixel al final. Tras el resultado del  $V_s$  obtenido, se procede a recorrer la secuencia de video para encontrar el momento en que se produce el valor de cambio, pudiendo aproximararlo a través de la siguiente expresión:

$$T_s = ((V_s - V_1) \cdot (V_s - V_i)) < 0$$

Para, una vez obtenidos ambos valores, realizar el cálculo del valor que tomará el nuevo frame, se hará a través de:

$$F_n = V_1 \cdot (1 - T_s) + V_\infty \cdot T_s$$

Con los anteriores resultados obtenidos, se procede a realizar el mismo principio para los frames de la secuencia a color, ya que las anteriores aproximaciones fueron realizadas sobre escalas de grises.

El procedimiento se basa en la realización de una media desplazante que comienza con el almacenamiento mediante suma de los 15 primeros frames de la secuencia, que serán utilizados como base de la secuencia sintética.

Con esta base de frames almacenados, la secuencia de video vuelve a ser recorrida desde la siguiente posición (frame 16).

En este segundo recorrido de la secuencia, en cada iteración será añadido un nuevo frame y restado el primero de la secuencia almacenada, permitiendo calcular una media por cada iteración al realizar la división entre 15.

El resultado de esta media desplazante en cada iteración es el mostrado en la siguiente figura:



**Figura 4-13: Media desplazante**

El anterior procedimiento devolverá una secuencia de video sin el primer plano, y en condición de que el audio sea añadido. Este proceso será realizado por la función **stabilized.m**, que creará un nuevo objeto de video para añadir el audio de la secuencia original.

Para que la reproducción final del video sea correcta, el audio añadido debe ir en sincronía con el frame reproducido. Esto se lleva a cabo extrayendo la frecuencia de muestreo con la que fue grabado el video original y dividiéndola entre la tasa de frames por segundo, obteniendo una constante que permitirá fijar que parte del audio será insertada en cada iteración de la grabación del audio en la secuencia del video final.



## 5 Pruebas y resultados

---

Han sido realizadas pruebas con videos de diferentes tipologías, duraciones y escenarios que se detallan a continuación:

- **Video 1:** “*video\_mano.avi*”

Grabación de la escritura de duración 6s segundos con rotulador de color azul, capturando en el plano de grabación la mano que escribe desde realizando un trazo de escritura, se logra eliminar por completo la mano que escribe, y mantener todo el trazo escrito.

- **Video 2:** “*video\_tutoria\_corto.avi*”

Grabación de una tutoría de duración 48s con rotulador color negro, capturando en el plano de grabación al sujeto desde el inicio hasta el final de la tutoría lo que dificulta la completa eliminación del primer plano en la secuencia de video por el algoritmo, manteniendo el trazo escrito.

- **Video 3:** “*video\_tutoria.avi*”

Grabación de una tutoría de duración 2min 41s con rotulador color negro, capturando en el plano de grabación un inicio y final del plano de pizarra sin el sujeto, donde se logra eliminar por completo el sujeto de la pizarra, y mantener el trazo escrito.





# **6 Conclusiones y trabajo futuro**

---

## **6.1 Conclusiones**

Para conseguir que este proyecto de desarrollo y llevar a cabo la finalidad del mismo por diseñar un sistema capaz de extraer el primer plano de una secuencia de video grabada por una cámara web posicionada delante de la pizarra durante una tutoría realizada en cualquier sala que dispusiera del equipamiento necesario; se han realizado pruebas con los distintos archivos de video capturados para evaluar el rendimiento del algoritmo desarrollado, llegando a la conclusión de que al estar realizando tareas de procesado de video el hardware es muy importante, ya que dependiendo de los tamaños de archivos que se deseen manejar por las duraciones de las tutorías implicaran cargas computacionales en mayor o menor medida ya que las tareas de estabilización y filtrado trabajan con dimensiones de imagen de 1920 x 1080 pixeles para lograr unos mejores resultados.

Tras aplicar el preprocesamiento para lograr la completa eliminación del FG en las secuencias de video, y el posterior posprocesado para lograr una secuencia de video con el audio en sincronía, es posible concretar que es viable sustituir el hardware especializado y de alto coste requerido en DTUSH, por la utilización de una cámara web de bajo coste que realice la grabación de una tutoría, obteniendo resultados satisfactorios.

Todo este proyecto alcanza la finalidad establecida en el mismo, abaratando los costes de un sistema anteriormente muy costoso y dando una nueva herramienta a tutores y estudiantes.

## **6.2 Trabajo futuro**

Tras los resultados obtenidos a lo largo de este trabajo, podríamos concretar que los esfuerzos futuros podrían ir destinados a conseguir la estructura de datos hipermedia, permitiendo el tratamiento de contenido de manera específica, esto permitiría llevar a cabo procedimientos OCR para completar el material hipermedia obtenido.

También sería interesante exportar los desarrollos implementados a otros lenguajes de programación, siendo de especial interés, la creación de una aplicación que permitiese que las grabaciones de estas tutorías fueran realizadas desde un smartphone o tablet, que como se indicó en el capítulo 4 la estabilización de vídeo requerida para compensar los movimientos de la cámara ya está realizada en este trabajo.



## Referencias

---

- [1] M. F. A. Tamayo, N. M. Martínez “Didáctica de la tutoría universitaria con soporte hipermedia”, *Inventio*. Vol. 13, Núm. 29. marzo 2017
- [2] Lim, Jae S., *Two-Dimensional Signal and Image Processing*, Englewood Cliffs, NJ, Prentice Hall, 1990, pp. 469-476
- [3] R. C. González, R. E. Woods, "*Digital Image Processing (3rd Edition)*", Prentice Hall, 2008.
- [4] T. Lindeberg, “Scale-space theory in computer vision”. Springer Science & Business Media, 1993.
- [5] A.K. Jain, "*Fundamentals of Digital Image Processing*", Prentice Hall, 2005.
- [6] R. C. Gonzalez, "*Digital Image Processing using MATLAB*", Gatesmark, 2009.
- [7] Chris Solomon, “*Fundamentals of digital image processing a practical approach with examples in Matlab*”, Willey 2011.



## Glosario

---

DTUSH	Didáctica Tutoría Universitaria con Soporte Hipermedia
FG	Foreground
MOOCS	Massive Online Open Courses
OCR	Optical Character Recognition
$V_s$	Switch Value
$V_1$	Valor primer pixel
$V_\infty$	Valor último pixel
$T_S$	Switch Time
$V_i$	Valor del pixel en la iteración
$F_n$	New frame

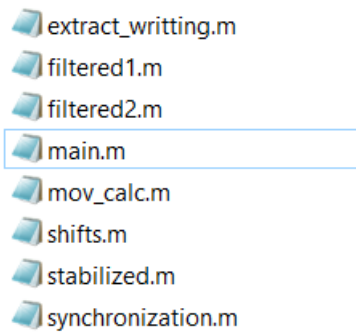


## Anexos

---

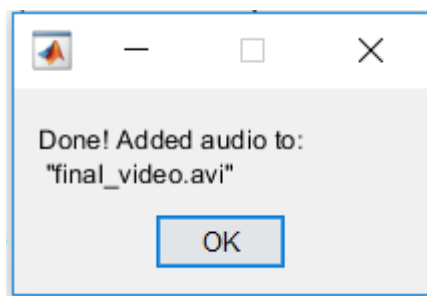
### A Manual de instalación

El programa desarrollado está comprimido en el archivo “*Apuntes\_pizarra.zip*” que contiene los archivos:



Para el correcto funcionamiento del programa el archivo “*Apuntes\_pizarra.zip*” ha de ser descomprimido y el video a procesar ha de estar ubicado en el mismo directorio del archivo *main.m*, de no ser así, deberá ser modificada la variable “file\_dir” con la ruta de ubicación del archivo de video.

La ejecución debe ser realizada desde el archivo *main.m*, una vez realizado todos los procesos será mostrado un cuadro de dialogo donde será mostrado la ubicación del archivo creado, que deberá cerrar para finalizar por completo la ejecución.







## ***B Manual del programador***

La descripción de cada función está descrita en las cabeceras de cada función donde aparecen los argumentos de salida y entrada a las mismas.

Para la descripción más detallada de la sintaxis de las llamadas a funciones ver detalles en Capítulo 3 – Desarrollo.

Los otros módulos usados propietarios de la herramienta son auto-explicativos y utilizan el comando “help” de Matlab.

Para obtener más información de uso ver:

<https://www.mathworks.com/>