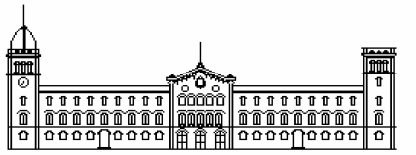


Treball fi de carrera

ENGINYERIA TÈCNICA EN INFORMÀTICA DE SISTEMES

Facultat de Matemàtiques
Universitat de Barcelona

REPRODUCCIÓN DE ACCIONES EN VÍDEO

Víctor Ponce López

Director: Sergio Escalera Guerrero
Realitzat a: Departament de Matemàtica Aplicada i Anàlisi. UB

Barcelona, 4 de juliol de 2010

RESUMEN

La expresión y comunicación oral es una de las competencias más relevantes para la vida personal, académica, profesional y cívica [3]. Según la *American Society of Personnel Administrators [1]*, se considera que una buena capacidad de comunicación oral es importante tanto para la obtención de un puesto de trabajo, como para un buen rendimiento en el trabajo [2]. El principal objetivo de este proyecto es obtener una herramienta informática que sea capaz de obtener una serie de características de la persona a partir de la información audio-visual. La extracción de las características obtenidas a partir de la expresión oral y el lenguaje no verbal es algo de especial interés en el análisis de los factores psicológicos que aparecen en la persona para analizarlos, con la finalidad de mejorar la calidad de la comunicación oral en presentaciones, entrevistas de trabajo… siendo ésta la meta final del proyecto. Se ha creado una versión del sistema que analiza una grabación y otra que lo hace en tiempo real a través de una WebCam.

L'expressió i comunicació oral és una de les competències més rellevants per a la vida personal, acadèmica, professional i cívica [3]. Segons la *American Society of Personnel Administrators [1]*, es considera que una bona capacitat de comunicació oral és important, tant per a l'obtenció d'un lloc de treball, com per a un bon rendiment en el treball [2]. El principal objectiu d'aquest projecte és obtenir una eina informàtica que sigui capaç d'obtenir una sèrie de característiques de la persona a partir de la informació audiovisual. L'extracció de les característiques obtingudes a partir de l'expressió oral i el llenguatge no verbal té especial interès per l'anàlisi dels factors psicològics que apareixen en la persona per a analitzar-los, amb la finalitat de millorar la qualitat de la comunicació oral en presentacions, entrevistes de feina… sent aquesta la meta final del projecte. El sistema s'ha aplicat a 15 vídeos de projectes de final de carrera i presentacions d'alumnes de quart curs. S'ha creat una versió del sistema que analitza una gravació i una altra que ho fa en temps real a través d'una WebCam.

The oral expression and communication is one of the most important competences for personal, academic, professional and civic life [3]. According to the *American Society of Personnel Administrators [1]*, it is considered that a good oral communication skill is important for obtaining a job, and for a good efficiency at work [2]. The main objective of this project is to obtain a Software tool that is able to obtain a series of characteristics of the person from the audio-visual information. The extraction of the characteristics obtained from the oral and nonverbal language is something of particular interest in the analysis of psychological factors that appears in the person to analyze them, in order to get better the quality of oral communication: presentations, job interviews... this is the ultimate goal of the project. The system has been applied to 15 end career project videos and presentations of fourth course students. It has been created a version that analyzes a recording and other that makes it in real-time via WebCam.

ÍNDICE

1 - INTRODUCCIÓN

1.1 – Motivación

El estudio psicológico de la persona en las expresiones orales y su comunicación no verbal es muy importante en muchos ámbitos para cualificar el comportamiento humano de la persona. Existen métodos y actividades para que las personas mejoren estas capacidades y así estén convencidas sobre sus capacidades comunicativas [5,6], con el fin de obtener mejores resultados en sus presentaciones o en cualquier acto de comunicación oral.

Aunque está claro que aún hay que avanzar mucho en esta competencia, el sistema desarrollado permitirá a la persona ver el resultado de sus capacidades de comunicación oral con el fin de mejorarlas.

La fluidez en el habla, la mirada frontal o no frontal hacia el público, la posición y aceleración de las manos, la postura del individuo, las interrupciones en el habla, la voz temblorosa, el movimiento o rigidez de la persona, entre muchos otros factores que dan información sobre las aptitudes sociales del sujeto, son clave para su estudio en el sistema propuesto.

Se han realizado 30 filmaciones de defensas de proyectos de final de carrera y de presentaciones en alumnos de cuarto curso del grado de informática de la Universidad de Barcelona.

1.2 – Tecnologías actuales

Actualmente existen métodos automáticos para extraer estas características, basados en ropas especiales, sensores o colores específicos que permiten saber la posición y/o aceleración de las regiones de interés, como cabeza, brazos, manos…[7] El inconveniente de estos sistemas es la complejidad y los elementos artificiales que debe llevar la persona para llevar a cabo el estudio, ya que puede resultar aparatoso e incómodo. Teniendo en cuenta el estado de la persona por el hecho de estar en una presentación, entrevista o, en definitiva, un acto de expresión oral donde interviene (en la mayoría de casos) una cierta presión en el sujeto, si se añaden estos elementos artificiales, se puede producir un aumento de dicha presión que afecta a su acto de expresión oral [4]. Otros sistemas propuestos en [8,9] tienen el objetivo de trabajar en entornos no controlados, y se centran en la detección del color de la piel, contornos, movimientos, o extracción de fondo, lo que permite automatizar y dar más independencia tanto el sistema de reconocimiento como al sujeto que realiza las acciones.

1.3 – Propuesta de sistema

Con el sistema propuesto no se precisa de estos elementos artificiales. El sujeto puede ir tanto en manga larga como en manga corta, y únicamente se precisará del uso de una cámara, un micrófono y un computador con la aplicación. Esto facilita la forma en que se extraen las características de la persona y la cantidad de elementos artificiales necesarios para realizar el estudio, por tanto no supondrá tanta presión al individuo, en la mayoría de casos.

En cuanto las funciones principales del sistema, hay una primera fase del sistema que consta en la extracción de las características psicológicas del individuo que se han comentado anteriormente en el apartado 1.1. Estas características serán analizadas en la segunda fase para determinar estos factores que se estudian a partir de las características extraídas. Finalmente, los resultados serán avaluados por un clasificador, que ordenará estos factores por relevancia, lo cual permitirá hacer un

ranking y detectar las características más discriminales para nuestro análisis. Además, se obtiene una aproximación de las opiniones de los observadores, que correlacionan con las mejores características extraídas. En general, se ha comprobado que existe una correlación entre la calidad de los proyectos en cuanto a contenido y la defensa del mismo.

1.4– Estructura

En primer lugar, se hará la contextualización del problema desde un punto de vista analítico, con esquemas explicativos de la metodología usada y requisitos, tecnologías que se usarán, la planificación y los costes. En definitiva, en este apartado se hará una abstracción de lo que se quiere hacer, pero sin entrar en detalles.

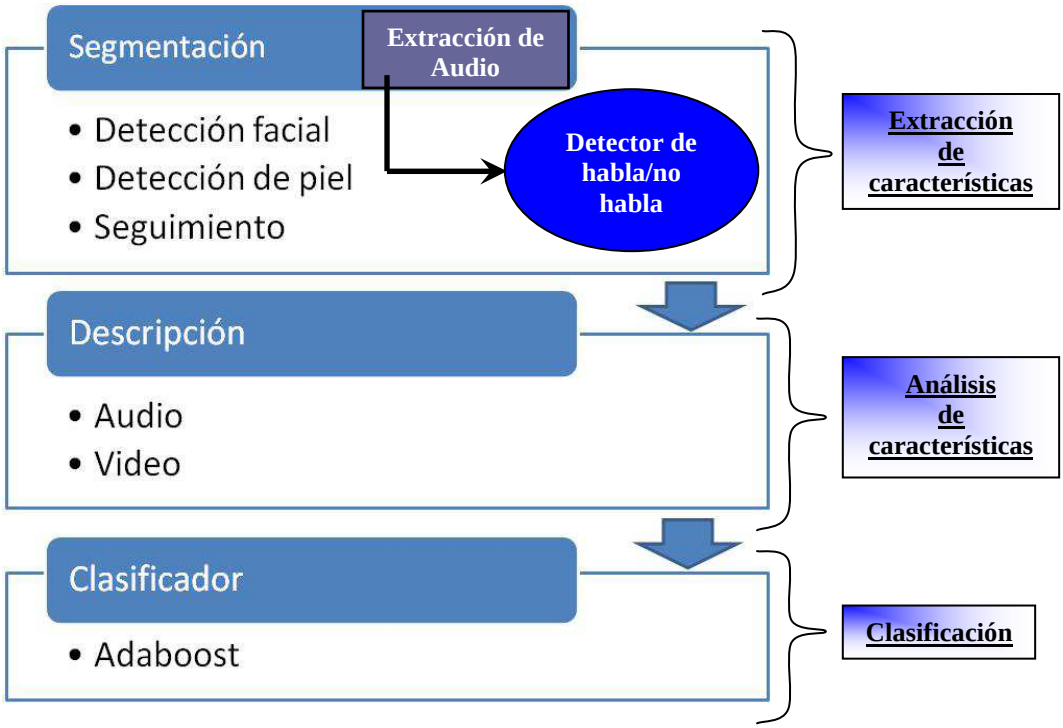
Posteriormente se explicará el diseño, profundizando en los detalles y explicando más las especificaciones y en cómo se quiere resolver el problema planteado, aplicando las tecnologías y metodología de la fase de análisis.

En tercer lugar, se mostrarán los resultados obtenidos, explicando las conclusiones según las hipótesis de las que partimos sobre lo que se desea obtener, y se mostrarán las referencias y agradecimientos que han ayudado a elaborar este sistema.

2 - ANÁLISIS

2.1 – Metodología

A continuación se describen brevemente las tres fases técnicas más generales y destacables de las que debemos tomar una primera referencia del sistema a desarrollar:



2.1.1 – Extracción de características

Dado que el sistema debe tener en cuenta el audio y/o el vídeo, pudiendo diferenciar las grabaciones con ausencia de audio, la extracción de características debe estar preparada para ambas cosas. Por tanto, se destacará la extracción de audio por una parte y la de vídeo por otra, donde ésta última hace referencia a la segmentación.

2.1.1a – Extracción de Audio

Para realizar la extracción de audio, se precisará de un **detector de habla/no habla**, es decir, un mecanismo que detecte en cada *frame* de la grabación si el sujeto está hablando o no y que guarde esta información.

2.1.1b – Segmentación

En cuanto a la extracción de características de vídeo, lo primero que se hará será aislar las zonas de interés a través de la segmentación, para obtener las posiciones relativas a las regiones de la cara y las extremidades superiores, que proporcionarán toda la información visual requerida. Para llevar a cabo este proceso, serán necesarias cuatro subfases que se deberán llevar a cabo en todo momento por el orden citado: **detección facial**, **detección del modelo de color de la piel**, **segmentación de color** y **seguimiento de las zonas de interés**.

2.1.1.1 – Detección facial

La detección facial será un primer objetivo, en el que se usarán métodos conocidos para detectar la cara de la persona. En el apartado de diseño se explicará con más profundidad el procedimiento y el algoritmo que se usará para llevar a cabo una detección facial teniendo en cuenta los posibles casos que pueden existir.

2.1.1.2 – Detección del modelo el color de la piel

Una vez realizada la detección facial realizada, se guardaran las posiciones pertenecientes a los píxeles donde se encuentra la cara. Esta posición permitirá seleccionar una porción de la cara que nos dará un modelo de color. El modelo de color será útil para poder detectar las regiones de los brazos [12], ya que son las otras zonas de interés. Esto crea una cierta discusión en cuanto a las posibilidades de tomar el modelo de color más apropiado según el escenario, que se explica a continuación.

Discusión

La iluminación, reflejos, movimientos… son factores que afectan directamente en el color. El escenario siempre está afectado por estos factores. Esto hace que las zonas cambien constantemente de color, ya que están en movimiento, lo que puede ocasionar una pérdida de las regiones según el color que se está tratando. En este caso, se debe dar un cierto tiempo para ver si se pueden recuperar las regiones, es decir, si pese al movimiento todavía el modelo de color tomado es válido para detectar las regiones. En el caso que no se recuperen las regiones en un cierto número de *frames*, se debe voler a tomar un nuevo modelo de color para esta nueva posición.

Otra forma que se ha propuesto es utilizar un modelo de color fijo creado a partir de diversas tomas en diferentes escenarios y diferentes personas. El inconveniente de esta forma es que el espectro de este modelo de color puede llegar a ser excesivamente amplio teniendo en cuenta todos los factores anteriores y otros más, como por ejemplo la raza de la persona.

Cualquier método de los dos hace una segmentación bastante correcta, pero hay que tener en cuenta que en ningún caso se consigue una segmentación perfecta, sobretodo en situaciones y escenarios complejos para el sistema.

2.1.1.3 – Segmentación del color

Una vez tomado el modelo de color con el procedimiento anterior, se llevará a cabo la segmentación de color respecto al modelo tomado. Para hacer esto se deseará obtener una imagen o *frame* en escala de grises donde destaquen todas las regiones de la imagen que encajen con este modelo de color, respecto la imagen o *frame* original.

2.1.1.4 – Detección de regiones de interés

Para llevar a cabo la detección de las zonas de interés, como extremidades superiores y la cabeza, se utilizarán las imágenes convertidas en escala de grises según el modelo de color tomado, agrupando las zonas con alta densidad de puntos que son candidatos de pertenecer a dichas regiones. Posteriormente se procede a la búsqueda y seguimiento de dichas zonas, guardando las posiciónes. Este procedimiento se repite para todos los *frames* del vídeo para detectar futuras regiones, teniendo en cuenta que éstas se desplazan suavemente en el tiempo, con el fin de realizar un proceso robusto de seguimiento de regiones. Con esto se conseguirá el objetivo de la primera fase de **extracción las características**, que se explicará con mayor detalle en la fase de diseño.

2.1.2 – Análisis de características – Descripción

Una vez detectadas las regiones de la cabeza y extremidades superiores con los métodos de segmentación y seguimiento descritos, se hace uso de las coordenadas de estas posiciones para extraer y definir un conjunto de descriptores que darán información sobre el comportamiento del sujeto. Además, se hará uso de la extracción de características de audio. Se han definido cuatro tipos de descriptores [14] que nos darán esta información: los **descriptores de actividad**, **descriptores de estrés**, **descriptores de involucración** y **descriptores de copia espejo**.

2.1.2.1 – Descriptores de actividad

Los descriptores de actividad vienen definidos por la densidad de habla del individuo en la totalidad de *frames* del vídeo. Se han diferenciado tres descriptores de actividad:

- *Habla*: Porcentaje de tiempo en el que ha estado hablando
- *No_habla*: Porcentaje de tiempo en el que no ha estado hablando.
- *Pausas*: Cantidad de intervalos superiores por cada dos segundos de pausa en el que no ha estado hablando, hasta un total de 20 segundos.

2.1.2.2 – Descriptores de estrés

Los descriptores de estrés vienen corresponden a la agitación corporal de los sujetos en el diálogo. Esta agitación se calculará a partir de la acumulación de distancias entre las coordenadas de las regiones en *frames* consecutivos. Se han diferenciado diez descriptores de estrés:

- *Agit_der*: Promedio de agitación del brazo derecho.
- *Agit_cab*: Promedio de agitación de la cabeza.
- *Agit_izq*: Promedio de agitación del brazo izquierdo.
- *Agit*: Promedio de agitación general (hay movimiento y actividad del sujeto).
- *Agit_direc_der*: Cantidad de desplazamiento realizado hacia la derecha.
- *Agit_direc_izq*: Cantidad de desplazamiento realizado hacia la izquierda.

- *Habla_agit*: Porcentaje de capturas donde hay alta agitación y se está hablando (énfasis).
- *No_habla_agit*: Porcentaje de capturas donde hay alta agitación y no está hablando (nerviosismo).
- *Habla_no_agit*: Porcentaje de capturas donde no hay agitación y se está hablando (calma).
- *No_habla_no_agit*: Porcentaje de capturas donde no hay agitación y tampoco se está hablando (no hace nada).

2.1.2.3 – Descriptores de involucración

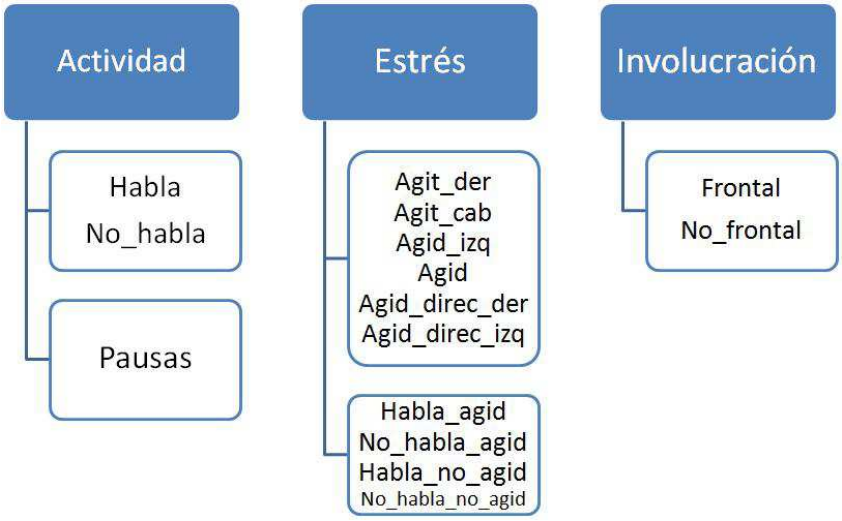
Los descriptores de involucración engloban las pautas de conducta que determinan si el sujeto está sumergido en el diálogo, teniendo en cuenta si el sujeto está dirigiéndose al público en su diálogo. Se han diferenciado dos descriptores de involucración:

- *Frontal*: Porcentaje de capturas frontales (aquellas en las que el sujeto mira al público/tribunal).
- *No_Frontal*: Porcentaje de capturas no frontales (aquellas en las que el sujeto no mira al público/tribunal).

2.1.2.4 – Descriptores de copia espejo

Definen la afinidad entre participantes de una conversación a partir de la imitación de gestos y pautas en el habla. En el caso particular de este sistema, el indicador copia espejo no aparece debido a que sólo hay una persona realizando la presentación.

Resumiendo de forma ilustrativa los descriptores de estrés que tendrá el sistema, se puede observar el siguiente esquema con la información sobre los descriptores psicológicos que se tendrán en cuenta:



2.1.3 – Clasificación

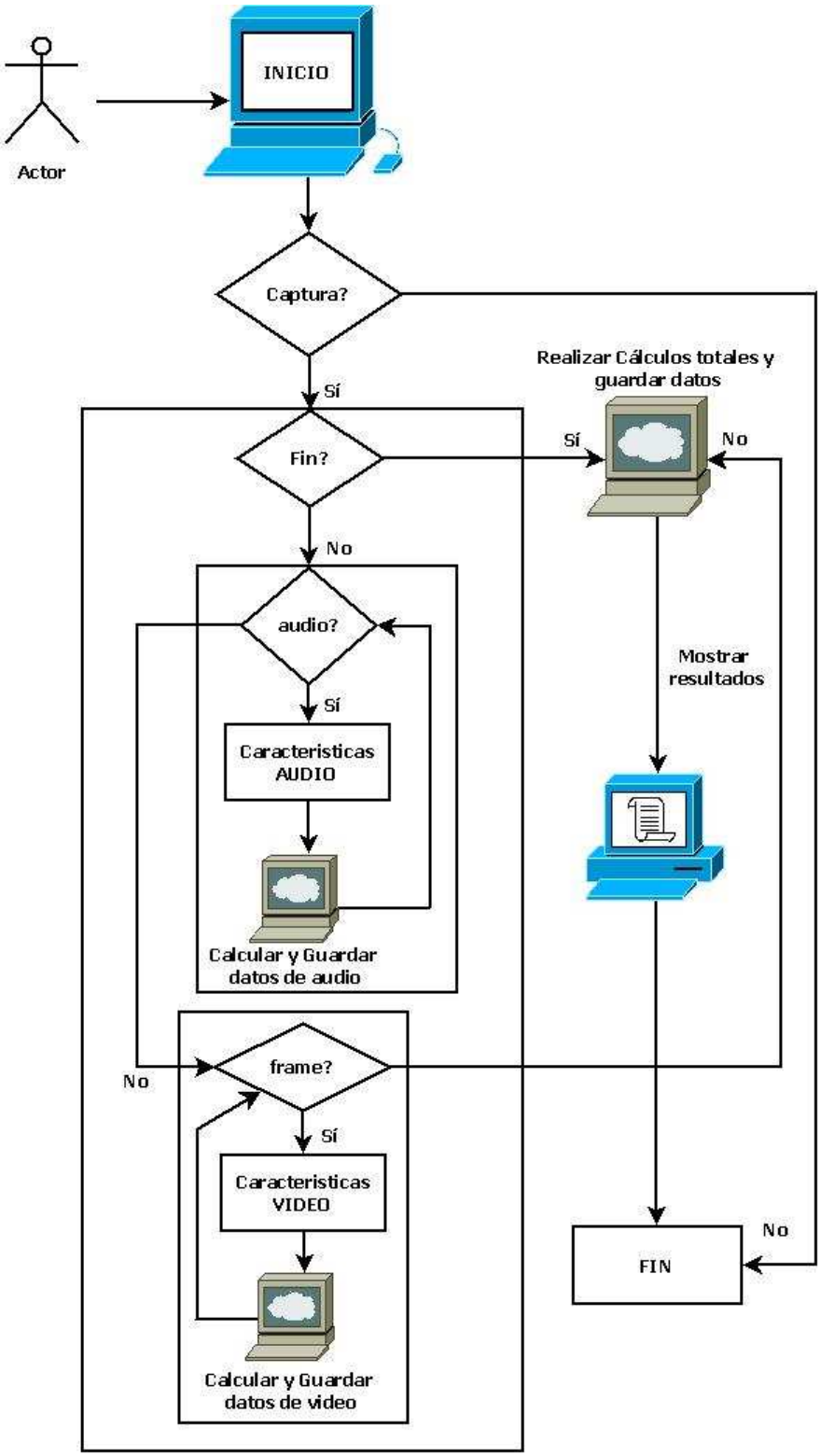
El objetivo de esta herramienta es extraer aquellos patrones que diferencien las presentaciones de mejor calidad de aquellas de menor calidad, así como la relevancia de cada una de ellas. Para ello, una vez que se han detectado, seguido, y descrito las características de cada sujeto en cada vídeo que corresponde a un acto de expresión oral, se hará uso de clasificadores estadísticos para analizar los datos respecto a la calidad de la presentación.

2.2 – Esquema del sistema

En este sistema hay que tener en cuenta que la interacción física del usuario con el sistema es mínima, es decir, desde un punto de vista de casos de uso, el usuario sólo debe ejecutar la aplicación, finalizarla cuando quiera (o cuando ésta acabe) y clasificar los datos obtenidos en el clasificador para obtener resultados, pero no se parte de un problema en el que usuario escoge múltiples opciones, sino que se trata de un sistema donde los métodos son automáticos. De esta forma, se podría decir que la interacción con el sistema es automática para el que se nombrará actor principal, ya que el propio sistema es el encargado de recoger y analizar "todo lo que ve". Además, el actor principal no tiene porqué ser el mismo que el que interactúa físicamente con la aplicación.

Dicho esto, en el diagrama de flujo general del sistema se deben diferenciar 2 grandes bloques: el de audio y el de vídeo. El funcionamiento general será que si se ha captado una captura, por ejemplo un fichero en formato de audio-vídeo o una captura vía cámara Web, en primer lugar se tratará el audio y en segundo lugar el vídeo. Hasta que no finalice el primer bloque no se pasará al segundo, exceptuando el caso en que no haya audio desde un primer instante, evento que hará que pase directamente al bloque de vídeo. En el caso de que no haya detección de vídeo desde un primer instante, será como si no se detectará captura y finalizará la aplicación. Cada bloque guardará en todo momento los datos mientras se esté en él, y una vez finalicen los 2 bloques o se de la orden de finalizar la aplicación, se realizarán los cálculos totales generales y se mostrarán los resultados. Acto seguido finalizará la aplicación. en el apartado el diseño se explicará con más profundidad, por eso hay que tener en cuenta que esto es una visión abstracta y muy general, pero cada bloque engloba tanto la extracción como la descripción, donde este segundo corresponde al almacenamiento de los datos en el diagrama.

En la página siguiente se puede observar el **diagrama de flujo** explicado.


Actor
INICIO
Captura?
Sí
Fin?
Realizar Cálculos totales y guardar datos
Sí
No
No
audio?
Sí
Caracteristicas AUDIO
Calcular y Guardar datos de audio
Mostrar resultados
No
frame?
Sí
Caracteristicas VIDEO
Calcular y Guardar datos de video
FIN
No

2.3 – Tecnologías usadas

Para el desarrollo del sistema se utilizará el lenguaje C/C++ y las librerías principales de *OpenCV* [15,16] usando sus funciones y métodos, aunque también se explicarán las tecnologías utilizadas para algunas funciones específicas del sistema.

La detección facial es un primer objetivo clave, como se ha podido observar en la metodología, por lo que se usará uno de los métodos actuales más extendidos: *Facedetect* por cascada de clasificadores de Viola & Jones [11]. Este método extrae un conjunto de características (Haar-like [11]) de imágenes con caras frontales y se aprende contra un conjunto de características de imágenes sin caras. Este clasificador es entonces probado sobre multitud de regiones de la imagen a diferentes escalas y posiciones. El resultado es la detección de regiones con alta probabilidad de contener una cara. En el diseño se explicará el algoritmo utilizado, que tendrá como propósito encontrar solamente una misma cara.

Para la detección del habla, lo primero que se hará es extraer el audio en formato .wav partiendo un fichero de audio-vídeo con un programa de extracción de sonido como *VirtualDub*, para utilizarlo posteriormente en el Software de [13], que permite crear un fichero de texto a partir de un fichero de audio, donde las columnas del fichero de texto corresponden a si el individuo está hablando o no, y cada fila corresponde a un *frame* de audio. La detección de sonido se hace a partir de los diversos canales de audio, que tratados a través de *Matlab* generarán este fichero de salida. Para más información consultar [13].

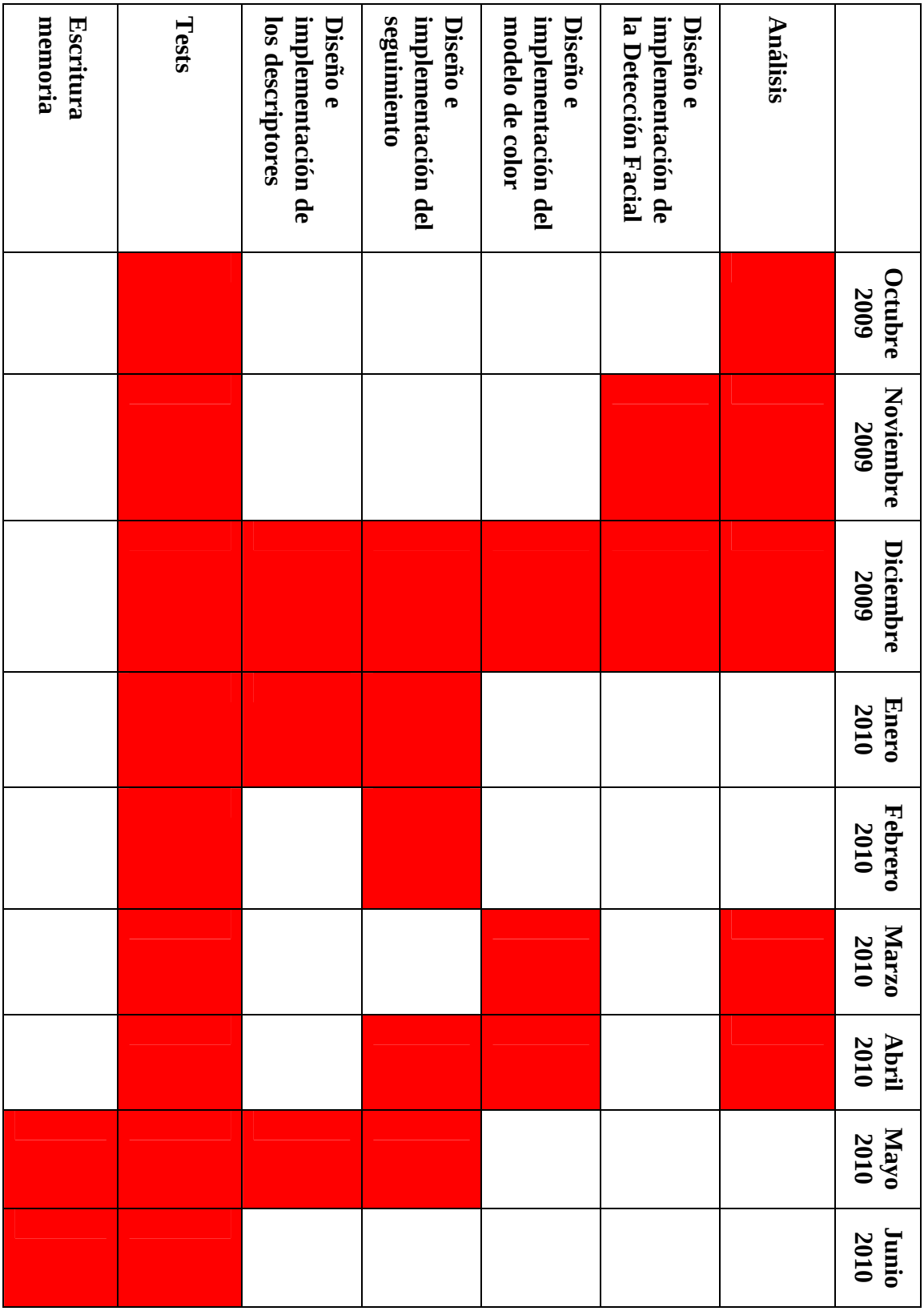
Como se ha comentado en el análisis, en la segmentación es necesario agrupar zonas con alta densidad de puntos que pueden ser candidatos de pertenecer a una región de interés. Para ello se utilizarán mecanismos de erosión y morfología de *OpenCV* combinados con las funciones de la librería *CBlobsLib* de *OpenCV* de [17], que permite extraer un conjunto de puntos, denominados *Blobs*, que determinan zonas independientes pertenecientes al modelo de color tomado. Con estos puntos o *Blobs* se puede trabajar y aplicar funciones de filtraje según el área, dibujar imágenes partiendo de estos *Blobs*, entre una gran variedad de funciones que serán altamente útiles en el desarrollo de este sistema.

Se utilizará *Adaboost* para el aprendizaje de un clasificador [10]. Utilizando *Adaboost* se obtiene un clasificador que combina distintas decisiones simples, basadas cada una sobre una única característica. Este método no sólo hace una selección de las hipótesis más relevantes, sino que además proporciona una regla de combinación basada en una suma ponderada de las características.

Detalles sobre este algoritmo se pueden encontrar en [10]. En la parte de evaluación del sistema, este método se usa para encontrar un clasificador que separe entre dos grupos principales de conversaciones, aquellas de mayor de aquellas de menor calidad. Además, también se utiliza para analizar el orden en el cual las características son seleccionadas de mayor a menor relevancia (*ranking*).

2.4 – Planificación

A continuación se mostrará un diagrama de Gantt para mostrar la planificación de cada una de las etapas del desarrollo del sistema.



2.5 – Costes

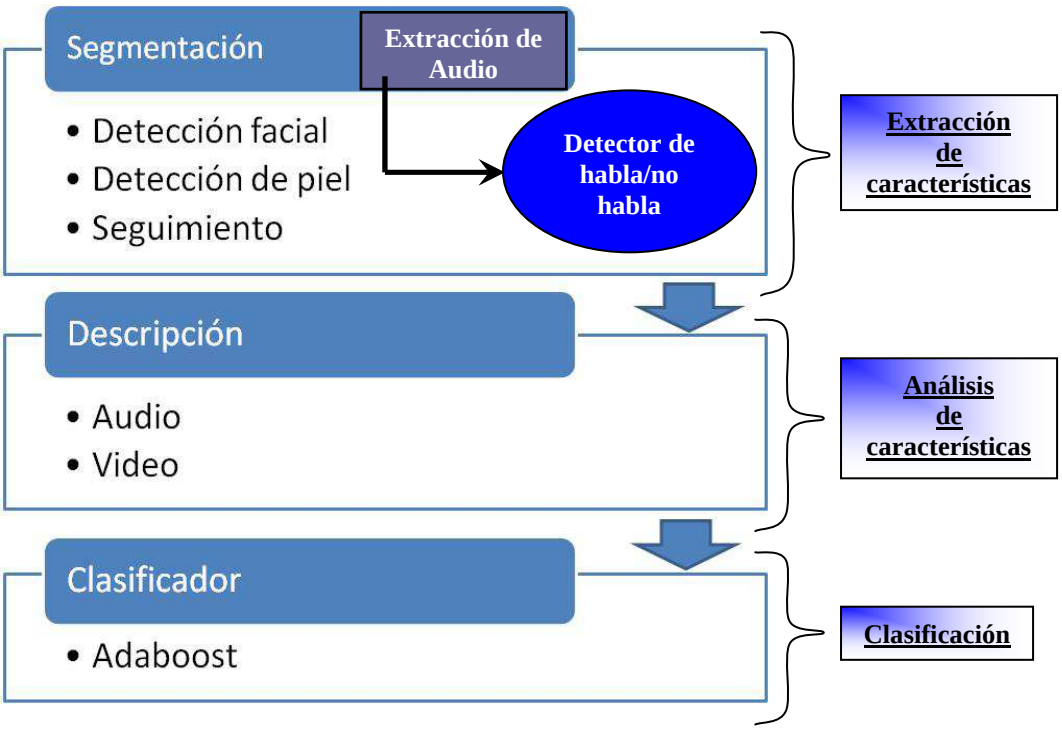
En la siguiente tabla se podrán observar los costes de la realización de este proyecto, desde su fase de análisis a su diseño, implementación y fase de testeo. También se tendrán en cuenta los materiales.

Concepto	Precio	Cantidad ó horas de trabajo	Total
Software (open source)	0,00	1	0,00
Ordenador	1200,00	1	1200,00
Cámara	80,00	1	80,00
hora de trabajo/programador	25,00	300	13500,00
Total			**14780,00**

3 – DISEÑO

Una vez contextualizado el problema, se procederá al diseño del sistema, explicando con más detalle cómo se solucionarán las partes explicadas en la fase de análisis. Hay que tener en cuenta que existen muchas formas de solucionar el problema planteado. La solución propuesta no se puede considerar la mejor, sino una forma más en la que se consiguen unos valores bastante buenos y donde hay un error inapreciable en lo que hace las fases de descripción y clasificación.

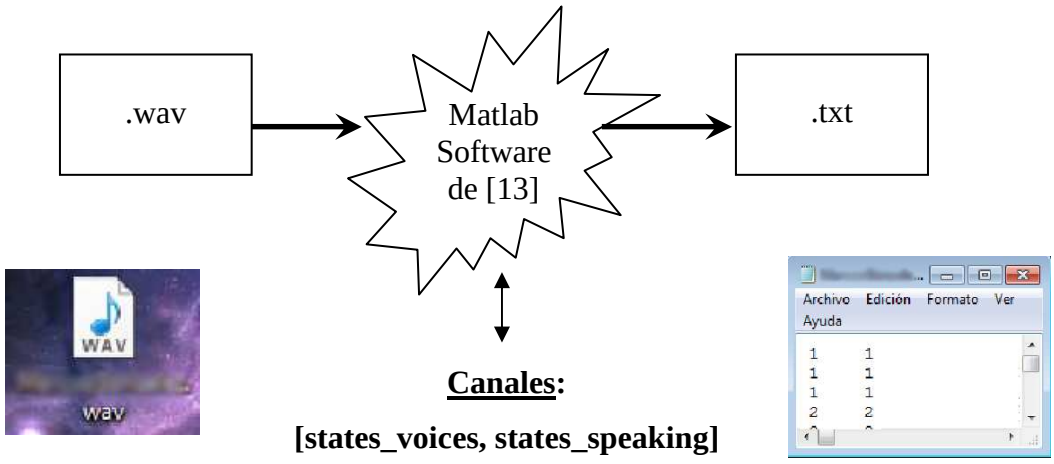
Según se ha visto y explicado en la estructura general del apartado [2.1], no estamos en un caso en el que tenemos una interacción física continua con el sistema, ya que el sistema es el responsable de extraer, analizar y clasificar los datos del actor principal, que hay que recordar que se ha definido como el actor del cual el sistema realiza el análisis. También se ha visto que se parte de dos grandes bloques de audio y de vídeo del cual extraer la información, que serán tratados con profundidad en este apartado. Dicho esto, se usará el esquema ilustrativo de la metodología, que se vuelve a mostrar a continuación, explicando cómo se ha resuelto cada etapa a través de pseudocódigo y diagramas.



3.1 – Características auditivas

En primer lugar y teniendo en cuenta la estructura del sistema y el esquema general del apartado [2.2], se explicará el primer bloque, pero centrándose solamente en la extracción de las características de audio.

Lo primero que se debe hacer para poder tener en cuenta el audio de un fichero de audio-vídeo es extraer dicho audio con un programa como el *VirtualDub*, tal como se comenta en el apartado [2.3]. Posteriormente se creará el fichero de texto con la información de audio desde *Matlab*, como se muestra en el siguiente diagrama:



Las variables *states_voices* y *states_speaking* dan información sobre si se está emitiendo sonido en el frame (1) o no (0), en el caso que se trate de sonido *stereo*.

Como ya se ha comentado en el apartado [2.3], las columnas del fichero de texto resultante dan información sobre si se habla o no, apareciendo un 2 caso de que hable Cada fila corresponderá a un *frame* de audio. Con esto ya se habrá hecho la extracción de audio**.**

3.2 – Características visuales

En segundo lugar y teniendo en cuenta la estructura del sistema y el esquema general del apartado [2.2], se explicará el segundo bloque, centrándose sólo en la extracción de las características de vídeo.

El orden de las fases de extracción de las características de vídeo deberá respetarse estrictamente según la estructura del sistema, ya que cada fase depende de la anterior.

3.2.1 – Detección facial

La tecnología usada se ha explicado en el apartado [2.3]. Pese a que la probabilidad de encontrar una cara con el Software comentado es bastante elevada, se necesitan tomar una serie de restricciones y variables para asegurar que sólo se detecta una misma cara, y que ésta es válida:

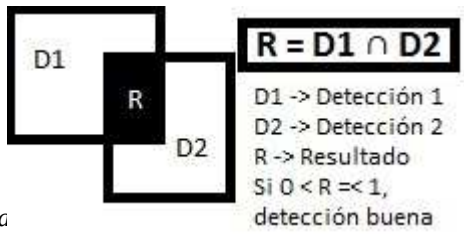
- Cálculo de la intersección para comprobar la proximidad entre detecciones.
- Número de capturas correctas, que cuando superen un mínimo darán por buena la detección.
- Número de detecciones erróneas, que dejarán un margen para dar por errónea la detección en caso de ya haber una detección correcta.

En la página siguiente se muestra el algoritmo en pseudocódigo:

Pseudocódigo *facedetect*:

Para toda cara detectada:

calcularInterseccion() ⟷



Si carasTotales > 1:

Si interseccion & radioCorrecto:

numCapturasCorrectas ← numCa

calcularNuevaInterseccion()

numCapturasTotales ← numCapturasTotales + 1

Si no se detectó cara:

Si numFallos <= minimFallos:

numDeteccionesFrontales ← numDeteccionesFrontales + 1

numFallos ← numFallos + 1

pintarCirculoConDatosAnteriores()

seleccionarPorcion()

Sino:

numCapturasCorrectas ← 0

detectado ← 0

numCapturasTotales ← numCapturasTotales + 1

Sino:

Si detectado & interseccion & radioCorrecto:

numDeteccionesFrontales ← numDeteccionesFrontales + 1

numFallos ← 0

pintarCirculoConDatosActuales()

seleccionarPorcion()

Sino, si detectado & (!interseccion || radioIncorrecto) & numFallos <= minimFallos:

numDeteccionesFrontales ← numDeteccionesFrontales + 1

numFallos ← numFallos + 1

pintarCirculoConDatosAnteriores()

seleccionarPorcion()

Sino si !detectado & No hay capturas mínimas correctas:

Si interseccion & radioCorrecto:

numCapturasCorrectas ← numCapturasCorrectas + 1

Sino si !detectado & Capturas mínimas correctas

Si interseccion & radioCorrecto:

numDeteccionesFrontales ← numDeteccionesFrontales + 1

numCapturasCorrectas ← 0

numFallos ← 0

detectado ← 1

pintarCirculoConDatosActuales();

seleccionarPorcion()

3.2.2 – Detección modelo de color

En el apartado [2.1.1.2] se discutía sobre la toma del modelo de color. En nuestro caso se ha utilizado el primer modelo discutido, por lo tanto este apartado y el anterior van íntimamente relacionados.

Se puede observar que en el algoritmo del pseudocódigo del *facedetect* hay una llamada a una función s*eleccionarPorcion()*. Esta función es la encargada de seleccionar una porción de la cara válida conseguida por el *facedetect*, usando los píxeles pertenecientes a las posiciones donde se encuentra, cuya porción será un rectángulo que se guardará y usará para tomar el modelo de color de la piel.

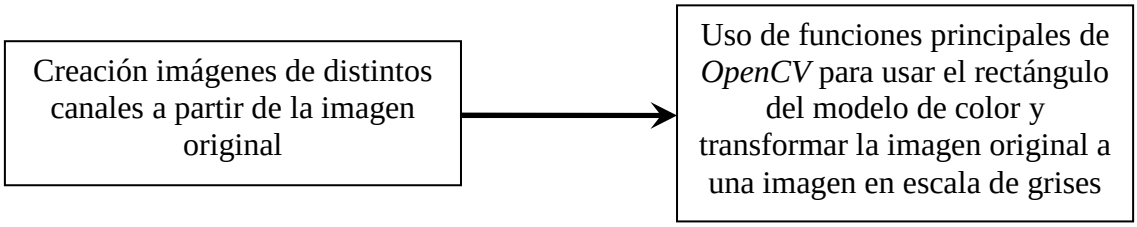
Como se ha comentado al principio de este subapartado, por el hecho de haber escogido la primera opción en la discusión sobre cómo tomar el modelo de color, será necesario una condición en la función *seleccionarPorcion()* que indique si es momento de realizar una selección o no. Por tanto, se usarán contadores para saber si ha habido *frames* suficientes como para considerar que se lleva bastante tiempo sin detectar las zonas de interés o si ha habido muchas pérdidas de las regiones de interés, y en caso de que así sea se ordenará tomar una nueva selección del modelo de color.

Aunque la implementación de esta fase es breve y no es necesario esquematizar ni explicar en pseudocódigo, se tiene que entender bien si se desea tomar un modelo de color robusto, complementando esta explicación con la del apartado [2.1.1.2] y conectándola con las fases posteriores.

Ejemplo modelo de color de la cara tomado

3.2.3 – Segmentación del color

A continuación se explicarán los pasos y funciones de *OpenCV* principales para realizar la segmentación del color:



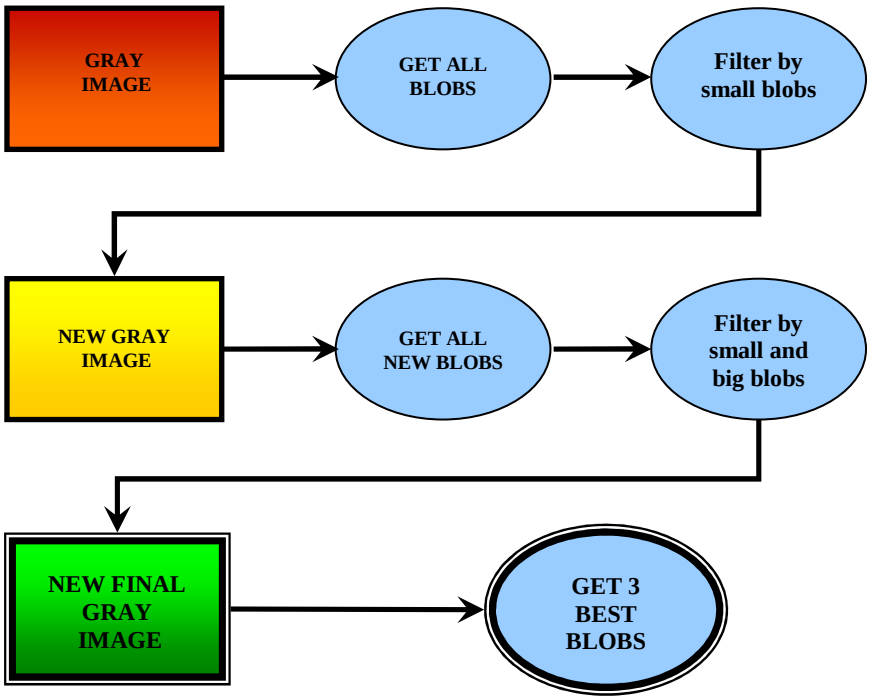
Frame	*cvCvtColor()*
Hsv	*cvSetImageROI()*
Mask	*cvCalcHist()*
Backproject	*cvResetImageROI()*
	cvCalcBackProject()
	cvCamShift()

Para más información sobre el algoritmo, consultar [16].

3.2.4 – Seguimiento

El seguimiento se divide en 2 fases: la aproximación de las zonas de interés y la búsqueda concreta de éstas.

En la aproximación de las zonas de interés lo que se hace es pasar una serie de filtros y restricciones usando la librería *CvBlobsLib*, como se había comentado en el apartado [2.3], que conllevará a determinar las zonas candidatas de pertenecer a las regiones de interés, como la de la cabeza o extremidades superiores. El algoritmo usado constará en coger los *blobs* de la imagen en escala de grises, filtrarlos según el área que tengan, excluyendo los que sean demasiado pequeños, aplicarles erosión y morfología para agrupar las zonas con alta densidad de *blobs* próximos entre ellos, y volver a filtrarlos según sus áreas, excluyendo de nuevo los que se consideren demasiado pequeños o demasiado grandes. Los *blobs* resultantes se pintarán en rojo. De estos últimos, se considerarán como buenos los 3 más grandes, que se pintarán en verde. Se hará uso de un contador que se irá incrementando si no se detectan *blobs* y hará tomar un nuevo modelo de color cuando pasen un cierto número de *frames* sin detección de *blobs*. A continuación se muestra el algoritmo con el siguiente diagrama:



La finalidad del algoritmo anterior es conseguir una nueva imagen en escala de grises con un número máximo de 3 blobs, de forma que el algoritmo de búsqueda y seguimiento de la segunda fase sea mucho más eficiente, rápido y concreto. Este algoritmo de búsqueda perteneciente a la segunda fase tendrá en cuenta los puntos siguientes:

- Una variable *marcarRegion* que indicará si se debe marcar una región con 3 zonas para realizar la búsqueda de las regiones de interés, tomando como referencia la cara. Esto se hará mediante la función *buscarRegionPrincipal()*, que se usará la primera vez que se deba realizar la búsqueda, o cuando el contador lo indique.
- El contador que indicará si se debe marcar la región o no será el *countMarcar*, que se incrementará cada vez que se pierda una región de interés y volverá a 0 cuando se detecten las 3 en un mismo frame. Si se llega a un máximo de pérdidas, se considerará que ha pasado demasiado tiempo desde que no se detectan las 3 regiones y por tanto habrá que marcar la región, complementando este evento con el punto anterior.
- 3 variables *detRegIzq*, *detRegCara* y *detRegDer* que se pondrán a 1 cuando se haya detectado la zona de la izquierda (brazo derecho), la de la cara o la de la derecha (brazo izquierdo).

En la página siguiente se muestra el algoritmo en pseudocódigo:

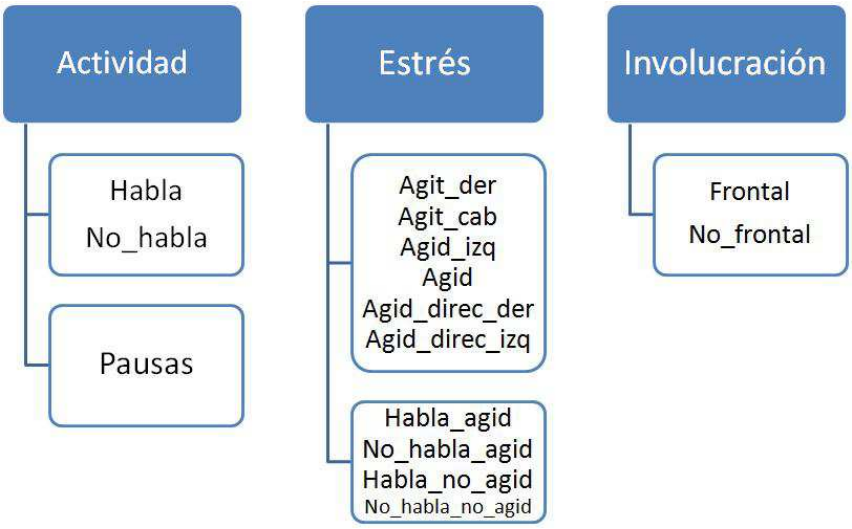
Pseudocódigo seguimiento:

Para todo blob:
Si no detectada región de la Cara (detRegCara):
Si hubo detección del facedetect & orden de marcarRegion:
Si el blobActual pertenece a la cara:
blobCara ← blobActual
detRegCara ← 1
Sino, si existe blobCara anterior & blobActual cercano al anterior :
Si el blobActual es independiente:
blobCara ← blobActual
detRegCara ← 1
Si no detectada región de la izquierda (detRegIzq):
Si orden de marcarRegion:
Si blobActual pertenece a la región izquierda:
detRegIzq ← 1
blobIzq ← blobActual

Sino, si blobActual supera los límites de la región:
Si blobActual pertenece a la región izquierda:
detRegDer ← 1
blobDer ← blobActual
Sino, si existe blobIzq anterior & blobActual cercano al anterior:
Si blobActual es independiente:
blobIzq ← blobActual
detRegIzq ← 1
Si no detectada región de la izquierda (detRegDer):
Si orden de marcar Región
Si blobActual pertenece a la región derecha:
detRegDer ← 1
blobDer ← blobActual
Sino, si blobActual supera los límites de la región:
Si blobActual pertenece a la región izquierda:
detRegIzq ← 1
blobIzq ← blobActual
Sino, si existe blobDer anterior & blobActual cercano al anterior:
Si blobActual es independiente:
blobDer ← blobActual
detRegDer ← 1

3.3 – Descriptores estadísticos

Recordando el esquema de los descriptores estadísticos del apartado [2.1.2]:



Se explicará cómo y en qué momento se elaborarán los descriptores de actividad, estrés e involucración.

Como se vio anteriormente, estos descriptores son cálculos estadísticos que, usando los datos obtenidos por la extracción de las características permiten obtener datos que, estudiándolos independientemente o combinándolos, dan información sobre los factores psicológicos de la persona.

En primer lugar y siguiendo el esquema general del apartado [2.2], se explicará el procedimiento del primer bloque, segundo bloque y parte final del diagrama de flujo una vez finalizada la extracción de características, momento en el que ya se pueden realizar los cálculos estadísticos y almacenar los datos.

3.3.1 – Descriptores de actividad

El cálculo estadístico de los descriptores de actividad, una vez extraídas las características de audio, se realizará mediante la lectura del fichero, incrementando un contador cada vez que se detecte un 2 en las columnas del fichero de texto, que indica que el sujeto está hablando, tal como se ha visto en el apartado [3.1]. Este valor, junto el número total de *frames* de audio, servirán para calcular el porcentaje de habla y no habla del individuo en la filmación una vez finalice la aplicación y se pase por tanto a la fase de realización de los cálculos totales generales. En este caso, los cálculos serán:

$$porcentHabla = \frac{habla \cdot 100}{framesAudio}$$

$$porcentNoHabla = \frac{(framesAudio - habla) \cdot 100}{framesAudio}$$

Estos valores se guardarán en un vector que dirá si en un determinado *frame* la persona estaba hablando, ya que esto nos servirá para las combinaciones de descriptores en los descriptores de estrés que se explicarán más adelante.

Como se quiere saber también el número de pausas superiores por cada 2 segundos, hasta un total de 20 segundos, se usarán los *Frames* Por Segundo (*fps*):

Hablando en términos de pseudocódigo:

Si $framesEnPausa >= x \cdot fps$, donde $x = segundos$ y 2 <= x <= 20 para cada x+2:

Se asigna estado del habla y se incrementa contador correspondiente a x

3.3.2 – Descriptores de estrés

Muchos de los descriptores de estrés se consiguen comparando las posiciones actuales y anteriores de las regiones de interés en cada momento.

Los cálculos de la agitación se harán en cada *frame* para ir acumulando los resultados y obtener valores promedios en todo momento. Esto se hará usando la función *calcularAgitacion()*, que se irá llamando en cada iteración y usará los valores de la agitación calculados anteriormente. El cálculo será el siguiente:

$$promAgitacion = promAgitacion \cdot (total - 1) + distCentros$$

El *promAgitacion* será el promedio de la agitación y se calculará tres veces, una para cada región, en caso de ser detectada. La variable *total* será el número total de *frames* donde hay agitación hacia una cierta dirección.

Para calcular el promedio de agitación general, se tendrán en cuenta los tres promedios de las tres regiones. De esta forma el cálculo será el siguiente:

$$promAgiGeneral = \frac{promAgiIzq + promAgiCara + promAgiDer}{3}$$

Este cálculo se realizará en la función *guardarPromedios()*, donde también se guardarán en un vector estos valores cada cierto número de *frames* para generar un histograma del promedio de agitación general cuando se finalice la aplicación. Además, se generará un vector con los valores promedios de la agitación general de cada *frame* para usarlo cuando se tengan que combinar diversos descriptores.

En los descriptores de desplazamiento de la agitación hacia una cierta dirección se usarán contadores que se tratarán en cada *frame* a través de la función *calcularAgitacion()*, donde se comprobarán las posiciones actual y anteriores de las regiones. Si son mayores que 0 se incrementará el total de desplazamientos hacia la derecha, y en caso contrario los de la izquierda. El porcentaje se realizará a partir del total de *frames* en los que existe desplazamiento de las 3 zonas a la vez cuando se de

la orden de finalizar la aplicación y se proceda al almacenamiento de los datos.

El promedio hará con el cálculo siguiente:

$$promedioDesplazamientoX = \frac{direccionX \cdot 100}{totalDesplazamientos}$$

donde X es la derecha o la izquierda

Para los descriptores que combinan habla y agitación, se usará el vector guardado en los descriptores de actividad, que dan información del habla del individuo en cada *frame* de audio, con el vector de agitación guardado para cada *frame* de vídeo. Los *frames* de audio y los de vídeo no son iguales, es decir, para una misma filmación, normalmente existen muchos más *frames* de audio que de vídeo, dependiendo de los formatos. En cualquier caso, habrá que establecer una relación para buscar la equivalencia de *frames* de audio con los de vídeo.

En la página siguiente se explica el algoritmo en pseudocódigo que incrementará los contadores que se usarán para los descriptores finales.

Pseudocódigo descriptores combinados:

Para todo frame de vídeo:

Calcular frame de audio equivalente (frameEquiv) al frame actual de vídeo

Si existe agitación en este frame & hay mucha agitación:

Si habla en este frameEquiv:

habla_agi ← habla_agi + 1

Sino:

noHabla_agi ← noHabla_agi + 1

Sino, si no existe agitación en este frame || no hay mucha agitación:

Si habla en este frameEquiv:

habla_noAgi ← habla_noAgi + 1

Sino:

noHabla_noAgi ← noHabla_noAgi + 1

Luego se calcularán los porcentajes con los contadores *habla_agi, noHabla_agi, habla_noAgi, noHabla_noAgi* a través del cálculo siguiente:

$$porcentajesDescCombinados = \frac{contador \cdot 100}{framesVideoTotales}$$

3.3.3 – Descriptores de involucración

En lo que hace los descriptores de involucración, lo que se quiere hacer es usar un contador en el *facedetect* que se incremente cuando se detecte una cara válida, lo que querrá decir que el sujeto mira frontalmente. Esto se puede ver en más detalle en el algoritmo en pseudocódigo definido en el apartado [3.2.1], donde se puede observar la variable *numDeteccionesFrontales*, que se refiere a este contador.

De esta forma, el porcentaje de capturas frontales y no frontales vendrá dado por las siguientes expresiones:

$$porcentajeFrontal = \frac{numDeteccionesFrontales \cdot 100}{framesVideoTotales}$$

$$porcentajeNoFrontal = \frac{(framesVideoMostrados - numDeteccionesFrontales) \cdot 100}{framesVideoTotales}$$

Con esto, se puede establecer la relación de capturas frontales respecto las no frontales, siempre que el número total *frames* sin detección frontal sea mayor que 0:

$$relacion = \frac{numDeccionesFrontales}{framesVideoTotales - numDeteccionesFrontales}$$

3.4 – Interfaz

El diseño que se utilizará para la interfaz de la aplicación se basa en tres partes o ventanas: la primera mostrará los resultados, la segunda mostrará la imagen en escala de grises y la tercera mostrará dos barras referentes a la agitación y la involucración.

En la primera ventana de resultados se marcarán con rectángulos las regiones de interés completamente agrupadas, para que se pueda ver en tiempo real el seguimiento. Existen otras zonas que se podrían dibujar, pero que se dejarán comentadas en el código. Estas zonas son el círculo del *facedetect* para marcar la cara, la región grande separada en tres zonas candidatas de pertenecer a las regiones de interés y el rectángulo que selecciona la porción del color de la cara. El motivo por el cual se dejarán dibujados solamente los rectángulos del seguimiento es básicamente para que no quede una interfaz demasiado cargada y confusa.

En la segunda ventana se mostrará la imagen actual en tiempo real convertida en escala de grises y los *blobs* rojos y verdes candidatos de pertenecer a las regiones de interés, según se explica en el apartado [3.2.4].

La tercera ventana mostrará en tiempo real dos barras que se van moviendo cada cierto número de *frames*. Estas barras muestran los valores de los porcentajes de agitación y mirada frontal, mostrando en rojo lo que “no es bueno”, por ejemplo un porcentaje bajo de mirada frontal, y en verde lo que “sí está bien”, por ejemplo una buena involucración hacia el público. También se van guardando los valores de cada intervalo y se van acumulando, pudiendo ver en todo momento los porcentajes en las situaciones donde más se ha encontrado una barra.

La ilustración de la interfaz gráfica se podrá ver a continuación en el apartado donde se mostrarán los resultados obtenidos en las pruebas que se han realizado, planteadas al principio de este documento.

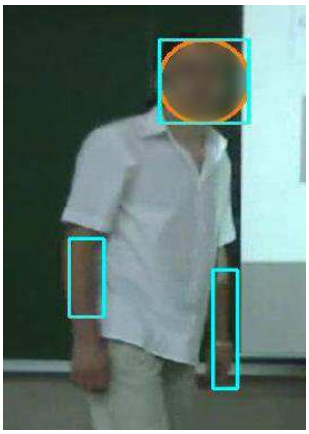
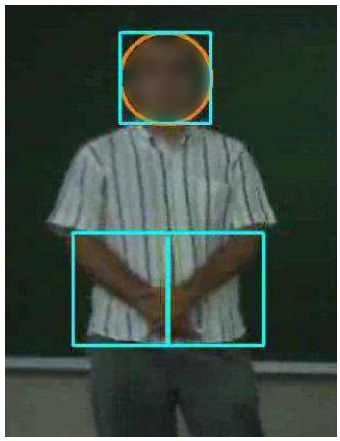
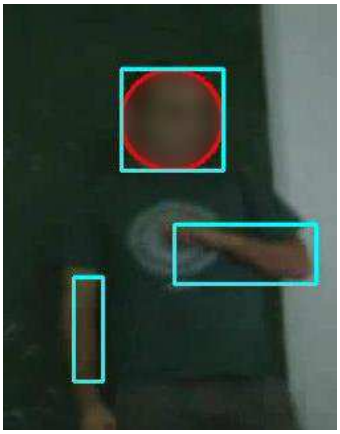
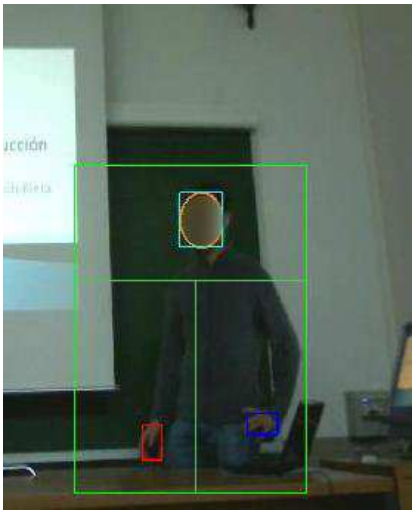
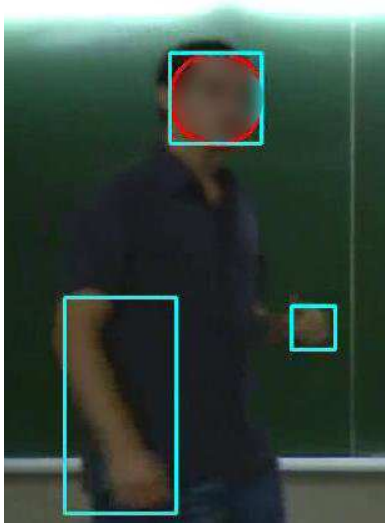
4 - RESULTADOS

En la planificación del proyecto del apartado [2.4], se ha podido observar que la fase de testeo está presente desde el primer momento, ya que hay que comprobar la validez de cada pequeña iteración implementada del sistema. Por lo tanto, esta fase de resultados sobre los tests realizados será bastante compleja y extensa.

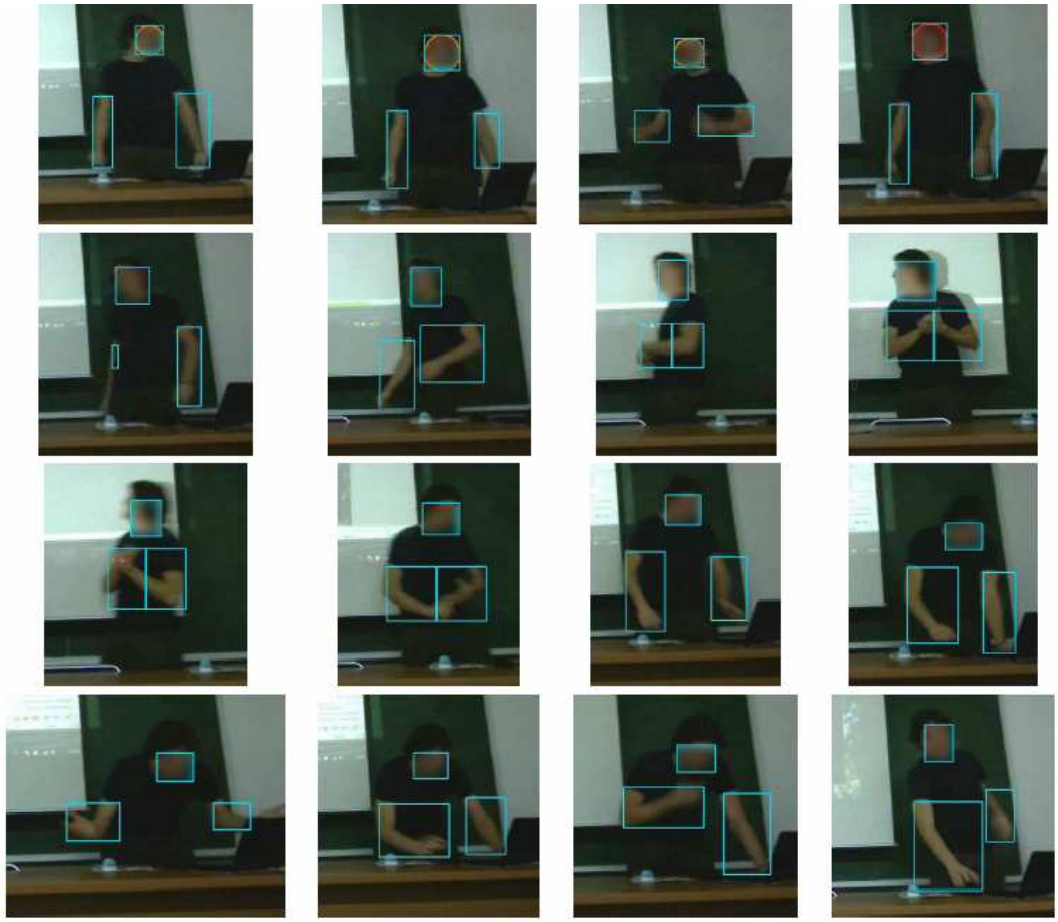
Como se había comentado en el apartado [1.1], se han realizado aproximadamente 30 filmaciones de defensas de proyectos de fin de carrera y de la asignatura de Visión Artificial de cuarto curso del Grado en Ingeniería Informática. Los vídeos han sido grabados con una misma WebCam a una resolución de 640x480 píxeles, y un *frame rate* de 25 imágenes por segundo. Además, todas las personas grabadas han signado un consentimiento de la filmación de sus proyectos para propósitos de investigación e innovación docente. Se puede observar un *frame* de las grabaciones en la siguiente imagen con las caras difuminadas, de acuerdo con el consentimiento:

Se puede observar que las grabaciones han sido filmadas frontalmente al sujeto respecto al tribunal, para así poder captar la desviación respecto la posición frontal y fijación de la mirada del sujeto. Esto es importante ya que todas las regiones de interés han sido normalizadas partiendo del área facial detectada con el objetivo de hacer comparables los valores de las características obtenidas por todas las personas. Estas zonas, como se ha explicado en los apartados anteriores, se detectan a partir de esta área facial, por lo tanto habrá que tener en cuenta la distancia del alumno respecto la cámara, porque el desplazamiento de los píxeles de las regiones de interés será mayor o menor en un caso u otro independientemente de la velocidad de agitación.

A continuación se muestran figuras ejemplo en diversos sujetos de las detecciones de regiones de interés, marcadas con rectángulos azules claro. Los círculos que rodean la cara significan que en ese *frame* el sistema hizo una detección facial y que por tanto se está mirando hacia el tribunal. Se puede observar también en una imagen un recuadro verde que separa las 3 zonas candidatas de pertenecer a una región de interés.

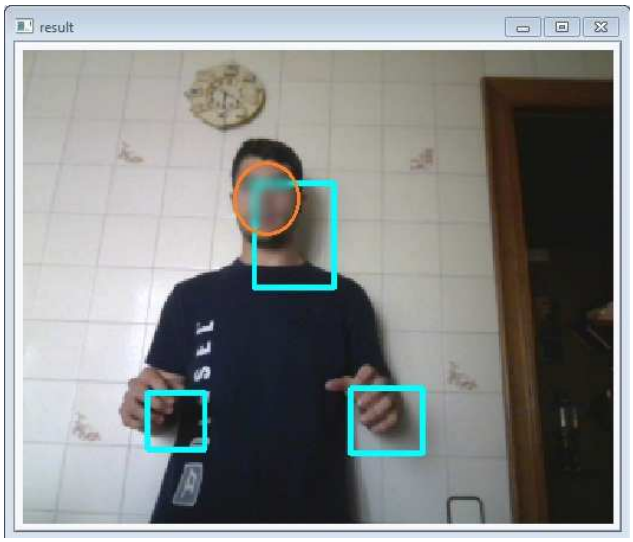






También se puede observar a continuación un conjunto de *frames* analizados correspondientes a una misma grabación sobre el mismo sujeto y ordenados cronológicamente.

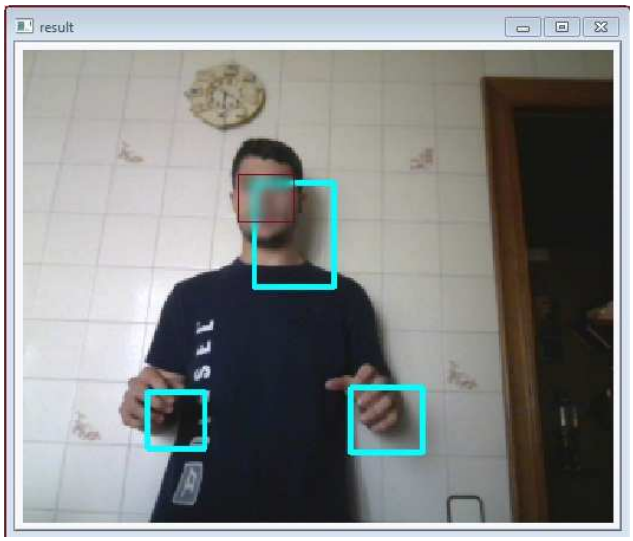


Para que se entienda y se vea de forma correlativa el procedimiento del análisis que realiza el sistema sobre un mismo *frame* y así entender cada una de las partes, se ha utilizado la versión de grabación directa desde WebCam, explicando con detalle y de forma ilustrativa las diversas fases.

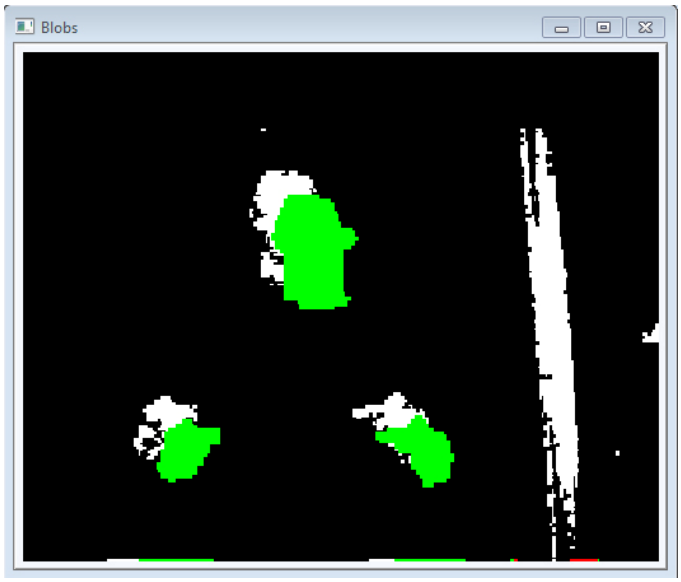
En la primera fase de detección facial comentada en el apartado [3.2.1], se puede observar el círculo naranja que rodea la cara y por tanto significa que el sistema ha realizado una detección válida, como se muestra en la siguiente imagen:



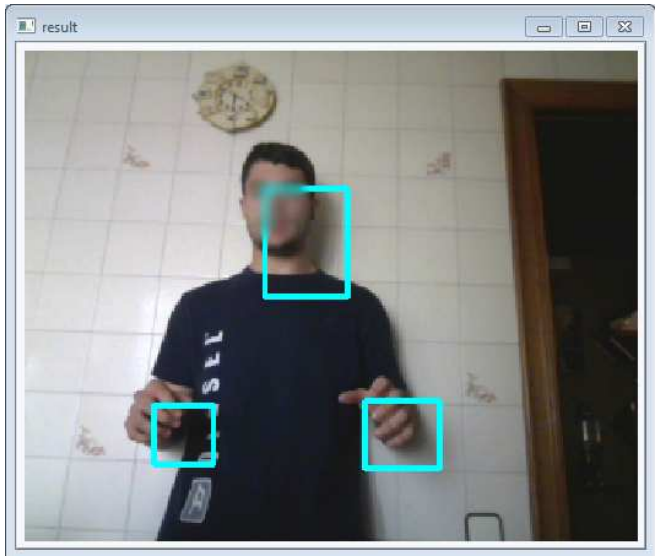
Posteriormente, se procede a la toma del modelo de color según lo comentado en el apartado [3.2.2], dibujando el rectángulo rojo de selección:



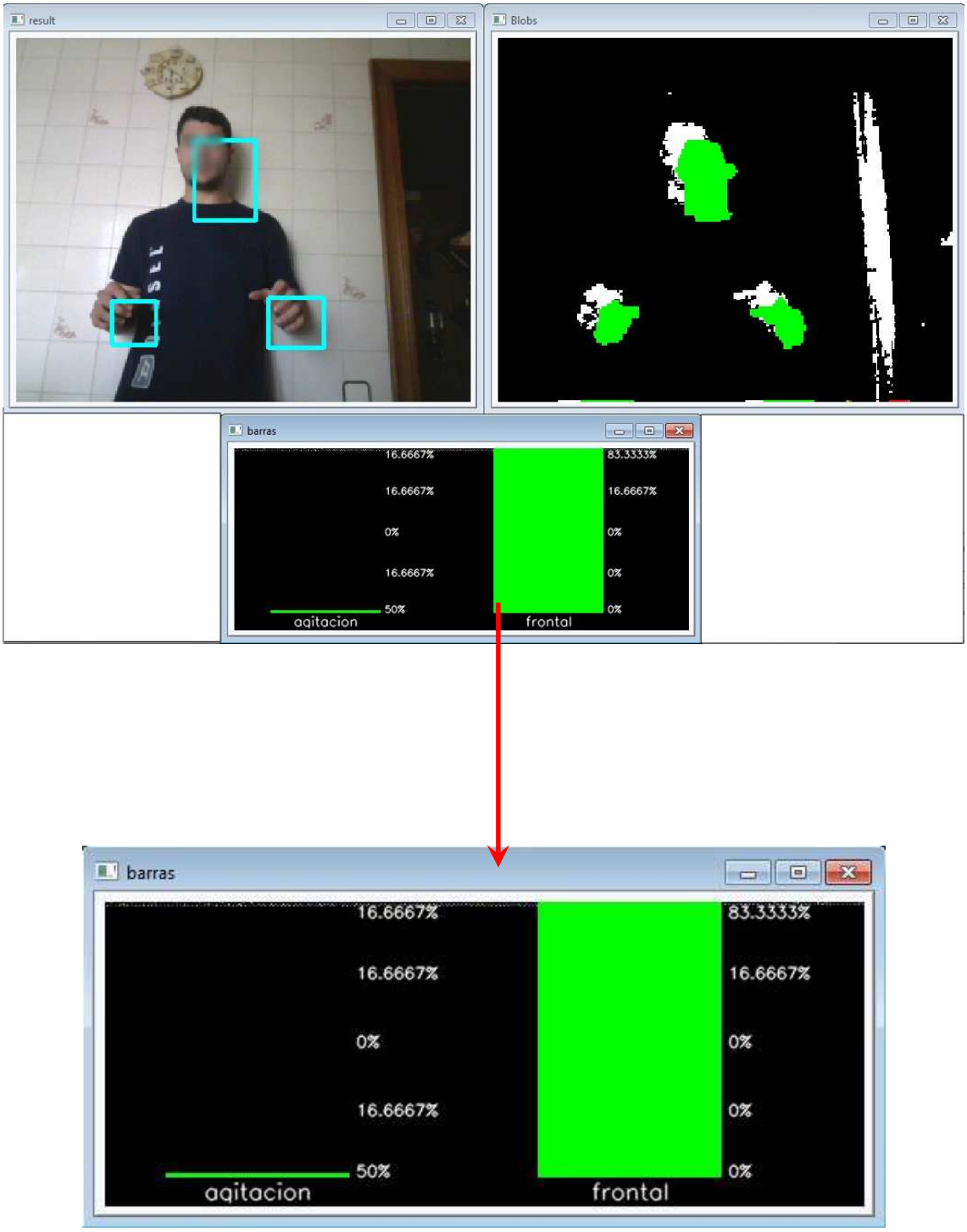
Según la discusión del apartado [2.1.1.2], también se detectará la zona de los ojos como modelo de color, cuya zona se deberá descartar ya que sólo interesa el color de la piel. Por lo que la ventana que muestra la segmentación de color una vez aplicados los algoritmos comentados en el apartado [3.2.3] quedará de la siguiente forma:



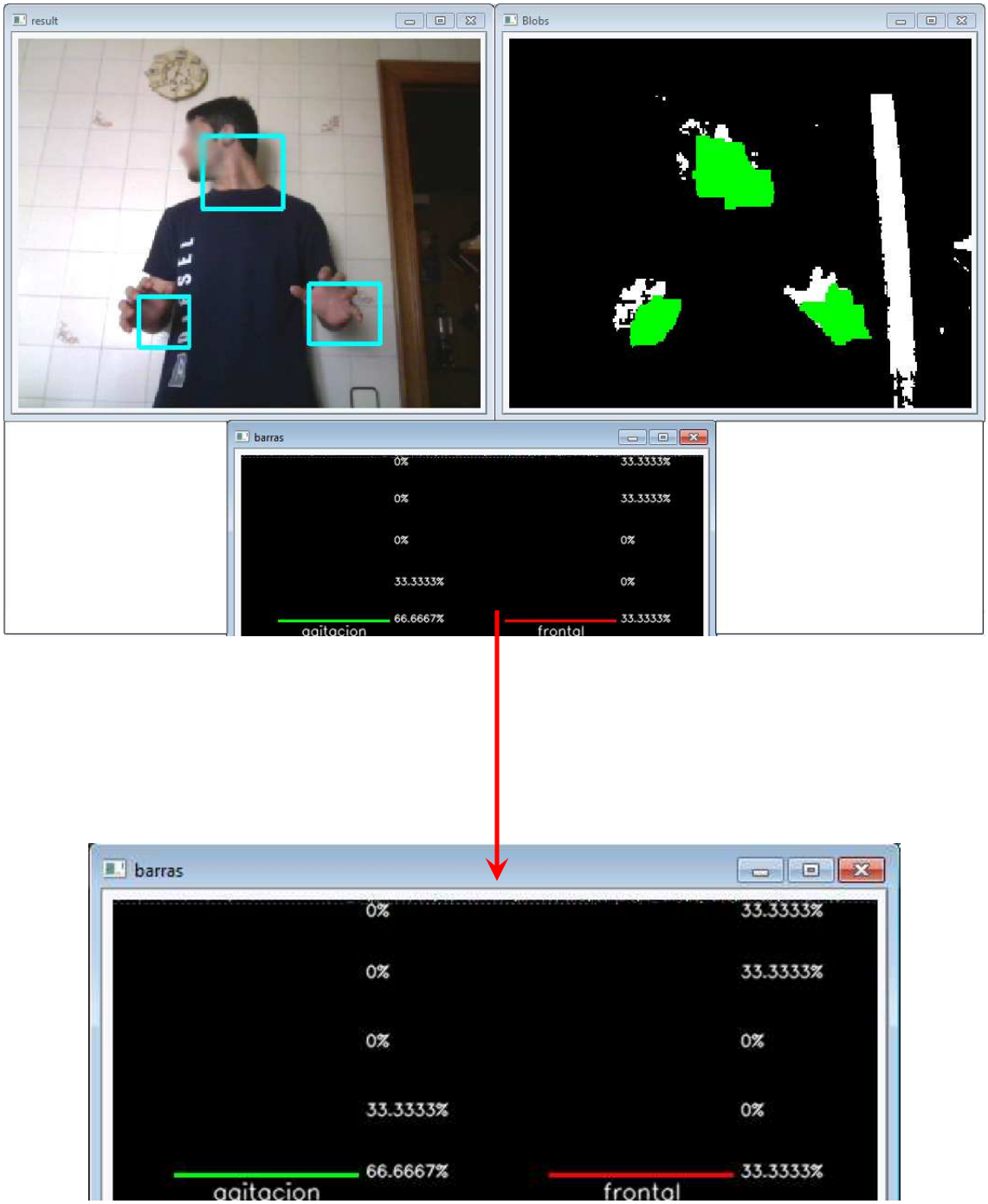
Se puede observar que pese a que en la imagen existe ruido, por ejemplo el marco de la puerta, las zonas en verde son las que interesan. No hay zonas rojas porque los *blobs* candidatos de pertenecer a las zonas de interés pertenecen realmente a las regiones, que se seguirán una vez aplicado el algoritmo del apartado [3.2.4], como se ha podido observar en las imágenes anteriores, y que ahora se vuelve a ilustrar sólo con las zonas de interés marcadas en rectángulos azules claro:



En lo que hace la interfaz del sistema hablada en el apartado [3.4], se comentaba la existencia de una ventana adicional con la información en tiempo real de la agitación y la involucración. A continuación se muestra la interfaz completa con un valor alto de la involucración (mirada frontal) para los *frames* tomados, y la ampliación de esta tercera ventana para visualizar con más detalle la relación entre el escenario y el estado de las barras:



A continuación se mostrará un contraejemplo, donde en el escenario se puede observar que no se está mirando frontalmente, y en las barras aparece corroborada esta información:



4.1 - Criterios de evaluación

Se han realizado dos tipos de evaluaciones. La primera consiste en encontrar aquellas características que mejor correlacionan las notas obtenidas por los alumnos con los patrones de comportamiento, como se ha visto en el ejemplo de clasificación anterior. Aunque esta nota final está influenciada por otros aspectos tales como la calidad del trabajo y la escritura de la memoria, se quiere analizar si existe una parte comunicativa relevante que influencia las calificaciones finales.

En la segunda evaluación se ha realizado una encuesta a 30 sujetos para que visualizaran y evaluaran la presentación de los alumnos. Con estos datos se pretende detectar si existe correlación entre las observaciones de los etiquetados calificando los vídeos, para posteriormente utilizar el sistema y extraer aquellas características que maximizan la correlación con la previa opinión de los observadores. Ambos experimentos se han realizado considerando problemas binarios, es decir, analizando las características que mejor separan entre dos grupos de presentaciones, que se podría decir que son las de mayor y menor calidad.

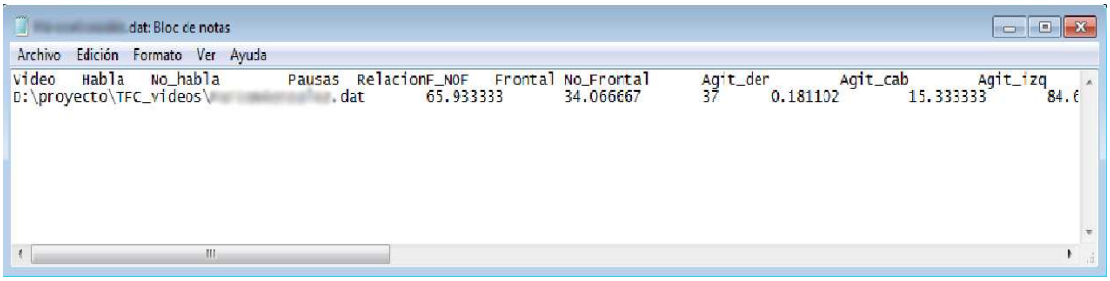
En cuanto el análisis de comunicación a partir de las calificaciones finales de las presentaciones, el clasificador primero determina cuál es la característica de mayor discriminabilidad, y ésta será la primera en el *ranking*. Seguidamente, los valores de estas características son extraídas de los datos, y se vuelve a lanzar el clasificador obteniendo la segunda característica de mayor discriminabilidad, y así sucesivamente. Se podrá observar que la mayoría se centran en la agitación del sujeto, el habla y la mirada frontal para clasificar las mejores presentaciones, mientras que la poca movilidad y paradas en el habla penaliza la presentación. En particular, el clasificador será capaz de separar correctamente dos particiones de 15 vídeos combinando información de las tres primeras características. Estas dos particiones serán de los 15 vídeos con nota más alta y los 15 vídeos con nota más baja.

Se ha realizado un análisis de comunicación a partir de las calificaciones obtenidas por un conjunto de observadores externos, donde la separación entre mejores y peores presentaciones se realiza a partir de la opinión de un conjunto de observadores. En particular, en las filmaciones se han mostrado a un conjunto de 30 sujetos

investigadores y docentes de la Universidad de Barcelona. Obviamente las notas a priori no son comparables ya que cada observador tiene diferentes niveles de rigurosidad en las evaluaciones. Por este motivo, en lugar de una nota numérica, se ha pedido a cada observador que ordenara de mejor a peor cada una de las presentaciones, obteniendo una medida homogénea entre observadores. A partir de los *rankings* individuales, se ha calculado el *ranking* promedio de cada filmación junto a su varianza. La primera observación interesante es que se pueden diferenciar claramente dos grupos de presentaciones a partir de su *ranking* (buenas y malas), a la vez que la varianza de los promedios es reducida, lo cual implica que existe un elevado acuerdo entre las anotaciones realizadas por los observadores. Además, comparando con las agrupaciones realizadas en el apartado anterior, sólo 2 de los 30 vídeos no encajan en la partición definida anteriormente. Esto nos indica que las opiniones de los observadores además están altamente relacionadas con las evaluaciones realizadas por los docentes que pusieron las notas reales de las presentaciones. Con el objetivo de analizar si las características principales cambian por la variación en las agrupaciones producidas por los dos vídeos que cambian de grupo, se ha realizado el mismo análisis que en el párrafo anterior. En el apartado siguiente se mostrará el ranking de las 8 primeras características que mejor separan las presentaciones con mejores *rankings* de las presentaciones con peores *rankings*, y cuáles son los valores obtenidos de cada una de ellas para realizar de forma correcta esta separación. En este caso, aunque el orden en el *ranking* ha variado, 7 de las 8 características siguen coincidiendo, dando más relevancia a las características de agitación en las primeras posiciones del *ranking*. En este experimento, el clasificador también es capaz de separar perfectamente las dos particiones de 15 vídeos combinando los valores de las tres primeras características del ranking.

4.2 – Clasificación

Una vez explicado el criterio de evaluación, se explicará qué sucede cuando finaliza la aplicación, momento en el que se genera un fichero *salida.dat* de este estilo:



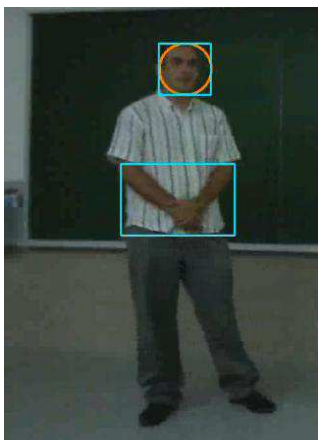
Estos ficheros contienen los valores de los descriptores que se han definido anteriormente de todos los vídeos, y se usarán para pasarlos al clasificador *Adaboost*, que se ejecutará 8 veces para que haga una selección más precisa de las 8 características de más relevancia comentadas en el apartado anterior [4.1]. El resultado de esta clasificación se puede observar en la figura siguiente.

Característica	Valor
Agit_cab	↑
Agit_direc_izq	↑
Agit_der	↓
Frontal	↑
Agit_izq	↑
Habla	↑
No_Habla_No_Agit	↑
No_Frontal	↓

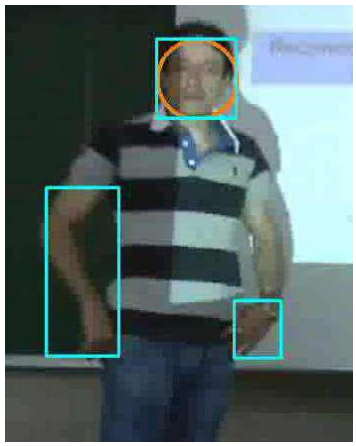
En la figura se muestra el *ranking* de características seleccionadas por el *Adaboost* para separar mejores notas de peores notas en función de las presentaciones y anotaciones de los observadores. A la derecha se muestra si los valores se seleccionan altos o bajos para discriminar las mejores notas.

4.3 – Discusión y propuestas de mejora

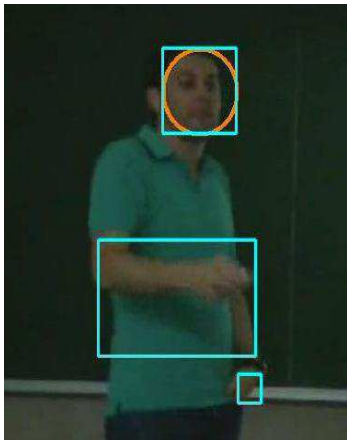
Los experimentos realizados muestran la viabilidad del sistema para extraer de forma robusta y automática patrones de comunicación útiles para la expresión oral y gestual de los alumnos. Aún quedan muchos puntos pendientes de ser analizados. En primer lugar, hay algunas situaciones en las cuales la segmentación no es del todo correcta. Estas situaciones se deben básicamente a la unión de las regiones y cambios debidos a la iluminación (a) y oclusiones (b, c), ya que hay un desplazamiento continuo durante todo el tiempo. Esto concuerda con la discusión del apartado [2.1.1.2], por lo que se mostrarán algunos ejemplos de ilustraciones sobre estos casos:



(a)



(b)



(c)

El siguiente paso consiste en depurar estas situaciones e incluir nuevos métodos más robustos que permitan una segmentación y seguimiento con mayor fiabilidad. Además, se podrá incluir información estructural y de expresión facial, tal y como determinar la orientación y estado de las manos, y no sólo su localización. El análisis más detallado de expresiones faciales también puede permitir añadir nuevas características que enriquezcan la descripción de los sujetos. También se quiere completar el análisis de audio. En esta versión se detecta el habla y no habla y se combina con información visual.

5 – CONCLUSIONES

Teniendo en cuenta los objetivos y finalidades comentados en el apartado [1.1] de este documento, que llevaban a la motivación del estudio que se ha realizado, se ha presentado una herramienta informática para el análisis automático de la comunicación oral y gestual de las personas para valorar sus aptitudes sociales en general, y para la defensa de proyectos, en particular. El sistema es capaz de detectar automáticamente las regiones correspondientes a cara, manos y brazos y extraer un conjunto de características que son analizadas mediante clasificadores estadísticos de Inteligencia Artificial y Aprendizaje Automático. Los resultados obtenidos sobre 30 filmaciones muestran la viabilidad y usabilidad del sistema para obtener valoraciones sobre la expresión oral y gestual de las personas, ofreciendo un "*feedback*" que tenga como finalidad tener en cuenta estas valoraciones para mejorar la calidad en los actos de expresión oral en los que se encuentran las personas a lo largo de su vida.

Un trabajo futuro a realizar que se ha comentado en los apartados anteriores de este documento con el fin de ampliar de este sistema consistiría, en primer lugar, en tener descriptores estructurales que estudiaran con más profundidad la forma y contornos las regiones en los sujetos. En segundo lugar, tratar en paralelo los bloques de audio y de vídeo para obtener información en tiempo real. También sería conveniente optimizar aún más las funciones y tratar los casos conflictivos que se han discutido. En último lugar, se considera la posibilidad de ampliar las funcionalidades de la versión *real time*.

6 – AGRADECIMIENTOS

Doy especialmente las gracias a todas las personas que han colaborado en el desarrollo de este sistema, desde la dirección del proyecto llevada por el Dr. Sergio Escalera, el soporte de Alberto Escudero, así como las personas que han dado su consentimiento para realizar las filmaciones. También agradecer al personal del Departamento de Investigación de la Facultad de Matemáticas de la Universidad de Barcelona, que han ayudado a testear el sistema y comentar sus sugerencias para mejorarlo.

7 - REFERENCIAS

[1] D.B. Curtis, J. L. Winsor, and R.D. Stephens. National preferences in business and communication education. Communication Education, Vol. 38 (1), pp. 6-14. 1989.

[2] J. L. Winsor, D.B. Curtis, and R.D. Stephens. National preferences in business and communication education: A survey update. Journal of the association of Comunication Administration, Vlo. 3, pp. 170-179. 1997.

[3] T. Allen, Charting a communicative pathway: Using assessment to guide curriculum development in a re-vitalized general education plan.. Communicative Education, 51(1) 26-39. 2002.

[4] S. Indra Devi and F. Shahnaz Feroz, Oral Communication Apprehension and Communicative competence among Electrical Engineering undergraduates in UTeM. Journal of Human Development and Technology, Vol. 1 Num. 1, June-December 2008.

[5] E. Valderrama, M. Rullán, F. Sánchez, J. Pons, F. Cores, and J.Bisbal, La evaluación de competencias en los Trabajos Fin de Estudios, XV JENUI, 2009.

[6] E. Valderrama et al. Guía para la evaluación de competencias en los trabajos de fin de grado y de máster en las Ingernierías, AQU Catalunya. 2009.

[7] J. Triesch and C. von der Malsburg, Robotic gesture recognition, Gesture Workshop, páginas 233-244, 1997.

[8] J. Martin, V. Devin and J. Crowley, Active hand tracking, Automatic Face and Gesture Recognition, páginas 573-578, 1998.

[9] F. Chen, C. Fu and C. Huang Hand gesture recognition using a real-time tracking method and Hidden Markov Models, Image and Video Computing, volumen 21, número 8, páginas 745-758, 2003.

[10] J. Friedman, T. Hastie and Robert Tibshirani, Additive Logistic Regression: a Statistical View of Boosting, Annals of Statistics, volumen 28, páginas 2000-2030, 1998.

[11] P. Viola and M. J. Jones, Robust Real-Time Face Detection, Int. J. Comput. Vision, volumen 57, número 2, páginas 137-154, 2004.

[12] M. Jones and J. Rehg, Statistical color models with application to skin detection, Intenational Journal of Computer Vision, volumen 46, páginas 81-96, 2002.

[13] http://groupmedia.media.mit.edu/data.php

[14] A. Pentland, Socially aware computation and communication, Computer, volumen 38, páginas 33-40, 2005.

[15] Introduction to IPL and OpenCV libraries. Bogdan Raducanu. Centre de Visiò per Computador

[16] OpenCV information and samples, http://opencv.willowgarage.com/wiki/

[17] Blobs library of OpenCV, http://opencv.willowgarage.com/wiki/cvblobsLib

APÉNDICE I

Este proyecto de Innovación Docente 2009PID-UB/04 ha sido realizado para el departamento de Aprendizaje e Investigación de la Facultad de Matemáticas de la Universidad de Barcelona, y ha conllevado a la obtención de una beca de colaboración en este departamento.

Las aplicaciones de este proyecto presentadas han sido las siguientes:

- JENUI 2010. XVI Jornadas de Enseñanza Universitaria de la Informática en Santiago de Compostela. http://www.jenui2010.es/

Métodos Automáticos para el Análisis de la Expresión Oral y Gestual en Proyectos Fin de Carrera

Resumen

La comunicación y expresión oral es una competencia de especial relevancia en el EEES. No obstante, en muchas enseñanzas superiores la puesta en práctica de esta competencia ha sido relegada principalmente a la presentación de proyectos fin de carrera. Dentro de un proyecto de innovación docente, se ha desarrollado una herramienta informática para la extracción de información objetiva para el análisis de la expresión oral y gestual de los alumnos. El objetivo es dar un "feedback" a los estudiantes que les permita mejorar la calidad de sus presentaciones. El prototipo inicial que se presenta en este trabajo permite extraer de forma automática información audio-visual y analizarla mediante técnicas de aprendizaje. El sistema ha sido aplicado a 15 proyectos fin de carrera y 15 exposiciones dentro de una asignatura de cuarto curso. Los resultados obtenidos muestran la viabilidad del sistema para sugerir factores que ayuden tanto en el éxito de la comunicación así como en los criterios de evaluación.

1. Motivación

Con la puesta en marcha de las titulaciones de Grado en el Espacio Europeo en Educación Superior, uno de los objetivos principales es que nuestro alumnado desarrolle una serie de competencias transversales y específicas de cada enseñanza.

La expresión y comunicación oral es una de las competencias más relevantes, considerándose un factor crítico para la vida personal, académica, profesional y cívica de los graduados [3]. En esta dirección, Curtis y Winsor constataron que la comunicación oral era el segundo factor más relevante para la *American Society of Personnel Administrators* [1], realizando posteriormente una encuesta a más de 1000 responsables de recursos humanos, llegando a la conclusión de que un buena capacidad de comunicación oral es importante tanto para la obtención de un puesto de trabajo como para un buen rendimiento en el trabajo [2].

En el caso particular de la Ingeniería Informática, el desarrollo de esta competencia ha estado básicamente relegada a la defensa de los proyectos fin de carrera. El listado y métodos de evaluación de las competencias específicas y transversales de un proyecto fin de estudios ha sido analizado y ampliamente discutido en el ámbito de las ingenierías, donde este tipo de actividades se viene desarrollando desde hace muchos años [5, 6]. En muchos casos, la defensa del proyecto fin de estudios era la primera ocasión en que el alumno se encontraba con la necesidad de comunicar sus resultados de forma oral, sin un entrenamiento previo. En [4] se hizo un estudio sobre el efecto de la aprensión y miedo a la presentación oral sobre la calificación obtenida por los estudiantes. Lo que se deriva de su trabajo es que la aprensión se traduce en peores resultados, y que cuanto más convencidos están los estudiantes sobre sus capacidades comunicativas, más cómodos se sienten y sus calificaciones son mejores. Para poder mejorar la percepción de los estudiantes sobre sus capacidades de comunicación, es necesario generar actividades que requieran comunicar conceptos y/o resultados, generando un buen "feedback"para que puedan ir mejorando sus capacidades.

Con la implantación del Grado en Informática en la Universidad de Barcelona, en algunas asignaturas se han comenzado a realizar pequeñas presentaciones por parte del alumnado

Característica	Valor
Agit_cab	↑
Agit_direc_izq	↑
Agit_der	↓
Frontal	↑
Agit_izq	↑
Habla	↑
No_Habla_No_Agit	↑
No_Frontal	↓

Cuadro 2: Ranking de características seleccionadas por el Adaboost para separar mejores notas de peores notas en función de las presentaciones y anotaciones de los observadores. A la derecha se muestra si los valores se seleccionan altos o bajos para discriminar las mejores notas.

Sería de interés, aunque no se analice el audio a nivel de palabra o nivel semántico, que se pueda además tener en cuenta las variaciones en la monotonía a partir del tono de voz.

Además de mejorar la parte correspondiente a la implementación del sistema, se plantea colaborar con psicólogos y otros especialistas en expresión gestual y verbal con tal de determinar un conjunto más concreto y exacto de características que hagan que el sistema se adapte mejor al diagnostico de presentaciones, así como pensar en diferentes escenarios donde esta metodología pueda servir también de utilidad para dar un "feedback" u obtener factores de calidad.

4. Conclusión

En este estudio hemos presentado una herramienta para el análisis automático de la comunicación oral y gestual de los alumnos de informática en la defensa de proyectos final de carrera. El sistema es capaz de detectar automáticamente las regiones correspondientes a cara, manos y brazos y extraer un conjunto de características que son analizadas mediante clasificadores estadísticos de Inteligencia Artificial y Aprendizaje Automático. Los resultados obtenidos sobre 30 filmaciones muestran la viabilidad y usabilidad del sistema para obtener valoraciones sobre de la expresión oral y gestual del alumnado, ofreciendo un "feedback" que permita mejorar la calidad de las presentaciones.

(a) (b) (c)

Figura 7: Ejemplos de detecciones imperfectas. (a) En algunos casos las manos y los brazos intersecionan, siendo difícil para el sistema artificial diferenciar entre las dos regiones que separan manos o brazos. (b) En los casos de cambios en iluminación, algunas partes de las regiones a detectar sufren un cambio brusco que los diferencia del modelo de color inicial, haciendo difícil la captura exacta de la totalidad de mano o brazo. (c) Un problema similar ocurre cuando una mano o brazo intersecciona con el opuesto y hay cambios debidos a cambios de iluminación u oclusiones. En este caso se produce que parte de un brazo se mezcle con parte del opuesto.

5. Agradecimientos

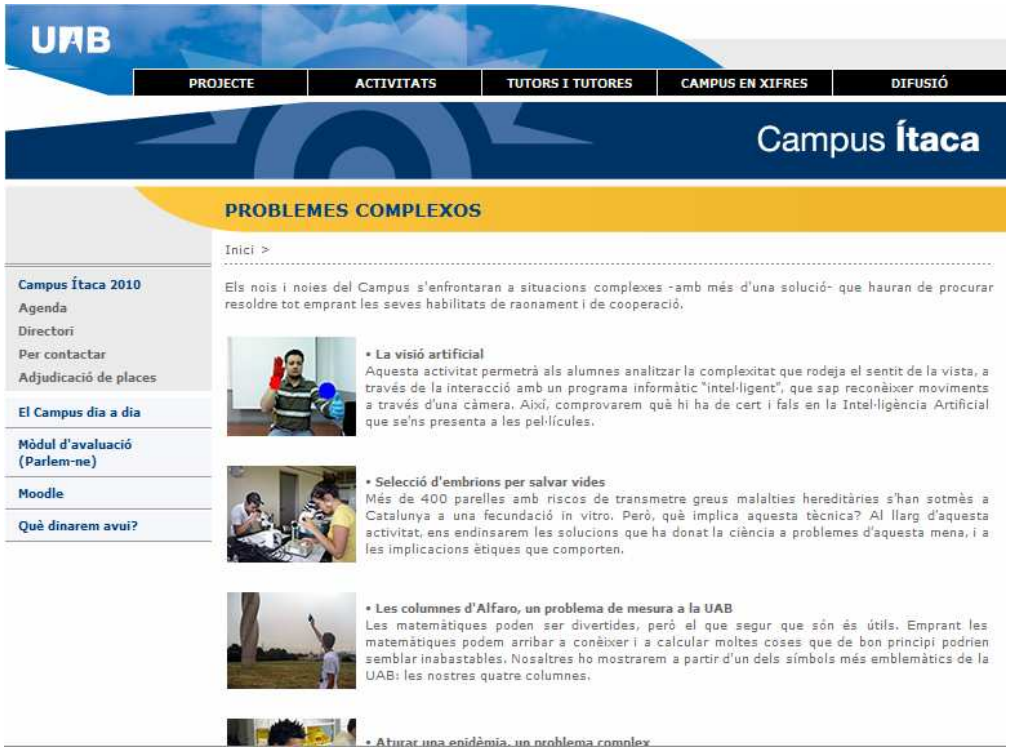
Este trabajo ha estado parcialmente financiado por el Proyecto de Innovación Docente 2009PID-UB/04 de la Universidad de Barcelona. Damos especialmente las gracias a Alberto Escudero y Víctor Ponce por su implicación en el desarrollo del sistema.

Referencias

[1] D.B. Curtis, J. L. Winsor, and R.D. Stephens. *National preferences in business and communication education.* Communication Education, Vol. 38 (1), pp. 6-14, 1989.

[2] J. L. Winsor, D.B. Curtis, and R.D. Stephens. *National preferences in business and communication education: A survey update.* Journal of the association of Comu-

- ITACA UAB. Jornadas de presentación de problemas complejos y actividades en el campus de la Universidad Autónoma de Barcelona. http://campusitaca.uab.cat/activitatespecprsimulacio.htm



APÉNDICE II

Junto a este documento se entrega un CD con el siguiente contenido:

- Documento PDF de la memoria.
- Códigos compilados y linkados, guardando en dos directorios con nombre "*app*" y "*appRealTime"* los ficheros ejecutables generados de las dos versiones del sistema y los ficheros necesarios para ejecutar la aplicación en cualquier máquina.
- Código fuente de las dos versiones del sistema.
- Fichero de texto *Readme.txt* con las instrucciones y usos de las dos versiones.