Regiones Salientes Complejas para Aplicaciones de Seguimiento Facial

Memoria del proyecto final de carrera correspondiente a la titulación de Ingeniería Superior Informática realizado por **Maximino Estévez Estévez** y dirigido por **Ágata Lapedriza** y codirigido por **Sergio Escalera** y **Xavier Baró**.

Bellaterra, 16 de junio de 2008

El firmante, Ágata Lapedriza, profesora del Departamento de Ciències de la Computació de la Universidad Autónoma de Barcelona

CERTIFICA:

Que la presente memoria ha sido realizada bajo su dirección por Maximino Estévez Estévez

Bellaterra, 16 de junio de 2008

Ágata Lapedriza

Agradecimientos

Quisiera empezar estas líneas agradeciendo a Ágata Lapedriza, directora de este proyecto, y a Sergio Escalera y Xavier Baró, codirectores de este proyecto, toda su ayuda y tiempo empleado para conseguir finalizar este proyecto.

También agradecer a Andreu Hidalgo, compañero de fatigas, el haber tenido que aguantarme durante todo este tiempo, todos esos buenos momentos y seguro que alguno malo, que no sólo hemos compartido durante el desarrollo del proyecto, sino en toda nuestra vivencia universitaria. Momentos que hemos compartido con otros compañeros, de los cuales espero no olvidar a nadie. Ingenieros o futuros ingenieros que hemos conocido en la ETSE (Carlos, Vanessa, Tania, Edgar, Pepelu, Gómez, Garru, Igual, Mario, los Guerrero, Miriam, Santi, etc.), integrantes del CVC, y otros que hemos conocido dentro y fuera de ella. A nuestros *Simeros*, que nos han aguantado tanto en este proyecto, aguantando tantas y tantas fotos; Mus, Marina, Grego, Janna, Alba, Tati, Noe, Irene, Nur, Natalie, Rocío, Brad y otros muchos que han pasado por nuestro querido SIM. Y muchos otros que aunque no aparezcáis aquí siempre tendréis un hueco en mis recuerdos.

Sin olvidarnos de nuestros compañeros de trabajo, esa cantidad de Ucanianos que han pasado por nuestro lado y siguen ahí, o han decidido cambiar de aires.

Tampoco olvidar a mi familia, en especial a mis padres, mi hermano y mi cuñada. Toda la que tengo a mi lado y otra mucha que sigue en Galicia porque así lo decidieron. Sin dejar de nombrar a mi abuelo, al que nunca olvidaré.

Y sin descuidar a Javi y a todos esos compañeros de andaduras veraniegas.

Y ante todo, pedir disculpas a todos los que por suerte o por desgracia aparecéis en alguna de las fotos de esta memoria.

Resumen

Este trabajo presenta una metodología para detectar y realizar el seguimiento de características faciales. En el primer paso del procedimiento se detectan caras mediante Adaboost con cascadas de clasificadores débiles. El segundo paso busca las características internas de la cara mediante el CSR, detectando zonas de interés. Una vez que estas características se capturan, un proceso de tracking basado en el descriptor SIFT, que hemos llamado pseudo-SIFT, es capaz de guardar información sobre la evolución de movimiento en las regiones detectadas. Además, un conjunto de datos públicos ha sido desarrollado con el propósito de compartirlo con otras investigaciones sobre detección, clasificación y tracking. Experimentos reales muestran la robustez de este trabajo y su adaptabilidad para trabajos futuros.

Resum

Aquest treball presenta una metodologia per detectar i realitzar el seguiment de característiques facials. En el primer pas del procediment es detecten cares mitjançant Adaboost amb cascades de classificadors dèbils. El segon pas, busca les característiques internes mitjançant el CSR, detectant zones d'interés. Una vegada que aquestes característiques són capturades, un procés de tracking basat en el descriptor SIFT, que hem anomenat Pseudo-SIFT, es capaç de guardar informació sobre l'evolució del moviment en les regions detectades. A més, un conjunt de dades públiques ha estat desenvolupat amb el propòsit de compartir-lo amb altres investigacions sobre detecció, classificació i tracking. Experiments reals mostren la robustessa d'aquest treball i la seva adaptibilitat per treballs futurs.

Abstract

This project is about a methodology to detect and track facial features. The first step of the procedure detects faces using Adaboost with a cascade of weak classifiers. The second step searches the internal face features using the CSR algorithm, detecting interest points. Once these features have been captured, a Pseudo-Sift process is able to save information about the movement's evolution of the detected regions. A data set has also been developed with the aim of sharing it with other detection, classification and tracking investigations. Real experiments show the robustness of this project and its adaptability for future works.

Índice

Índice de figuras

1. Introducción

La detección de caras es un trabajo de Visión por Computador que lleva cerca de 50 años en investigación, sobre el que se ha invertido gran esfuerzo debido a su amplitud de posibilidades prácticas. Actualmente podemos decir que la detección frontal de caras es un problema resuelto, ya que en la literatura podemos encontrar una gran amplitud de trabajos que ofrecen soluciones robustas a este problema. No obstante, cuando la detección facial se produce bajo diferentes condiciones, tales como cambios en el punto de vista, nos encontramos ante un problema no resuelto, donde tanto el seguimiento facial como su detección se presentan como tareas difíciles de procesar. En este trabajo se presenta una aproximación a dicho problema. El proyecto se basa principalmente en el seguimiento facial, el cual nos permitirá en todo momento tener información espacial de sus características, siendo invariante al punto de vista de la cámara.

Un sistema de reconocimiento facial es una aplicación dirigida por ordenador para identificar automáticamente a una persona en una imagen digital mediante la comparación de determinadas características faciales y una base de datos. Este reconocimiento es utilizado principalmente en Sistemas de Seguridad para el reconocimiento de usuarios. Consiste en un lector que define las características del rostro. De esta forma, cuando un usuario solicita acceso, sus características son verificadas contra la base de datos. Según la dinámica del sistema, éste podría llegar a ser poco fiable, ya que tanto la evolución de las características de nuestro rostro con el paso del tiempo como la gran variedad de rostros a diferenciar reducen la fiabilidad de los resultados. Por lo tanto, el objetivo principal de este proyecto se centra en la detección y seguimiento de caras. Para ello se han utilizado técnicas de Visión por Computador e Inteligencia Artificial que permiten el análisis de frames y secuencias de vídeo.

La mayoría de los trabajos de Visión basados en la detección de caras se usan para interacción hombre-máquina en robótica y en sistemas autónomos de vigilancia. El trabajo más robusto publicado recientemente sobre la detección de caras es el propuesto por Viola & Jones [1], que permite detectar caras frontales en frames a una velocidad cercana a tiempo real con una robustez elevada. El objetivo en este punto no ha sido tratar de hallar un método que supere la robustez del método comentado, sino que se ha utilizado dicho algoritmo como punto de partida de nuestro proyecto. Aquí hay que aclarar que el objetivo que se ha pretendido conseguir es seguir una cara detectada a partir de un frame con el algoritmo de detección de caras. La dificultad consiste en seguir la cara en una secuencia de video a medida que ésta cambia su apariencia, ya sea debido al cambio de ángulo, giros, oclusiones o deformaciones que se producen. Cabe destacar que el método anteriormente comentado de detección de caras ofrece gran fiabilidad de detección a través de un aprendizaje

estadístico basado en Adaboost, pero para que esta detección se produzca, la cara tiene que estar en estado frontal y totalmente alineada. De esta manera, y tal y como hemos comentado anteriormente, el seguimiento y detección de caras bajo diferentes condiciones en entornos no controlados aún sigue siendo hoy día un problema abierto.

La visión por computador es un campo de trabajo que nos ofrece posibilidades para tratar con el problema que estamos comentando (en el Anexo 8 se algunas de las aplicaciones más novedosas en la Visión Artificial). Un punto en común que tratan los diferentes problemas de la Visión por Computador es entender el funcionamiento del Sistema de Visión Humano (HVS) para tratar de analizar como éste resuelve problemas para los que nuestros ordenadores aún no están capacitados. Gracias al análisis del HVS, no sólo se está llegando a simular su comportamiento a través de medios electrónicos, sino que se está llegando al punto de poder alterar físicamente nuestro sistema visual. Cuando la retina está dañada o no funciona bien, los fotorreceptores dejan de funcionar, pero eso no quiere decir que toda la estructura del Sistema Visual Humano no pueda seguir funcionando. Por ello hay una parte de científicos que están desarrollando microchips de silicio que puedan dotar de Visión Artificial a aquellas personas a las que no les funcionan los fotorreceptores (ver anexo 8).

Volviendo a lo que nos centra, esta primera fase del proyecto es la detección facial. Se detecta la cara en un frame inicial del vídeo, a partir del cual se hará el seguimiento. Aunque se ha comentado que la detección facial es un problema resuelto y para ello se utiliza uno de los métodos más robustos que existen en este terreno, es importante mostrar el funcionamiento de estos algoritmos para apreciar la complejidad del proceso. Aunque una cara se encuentre en un estado alineado frontal, el algoritmo tiene que ser capaz de realizar la detección enfrentándose con cambios de escala, colores, expresiones, cambios de iluminación, etc, y a la vez evitando que se confunda un elemento del fondo de la imagen con una cara inexistente, lo que se conoce como un falso positivo.

Para la solución de este problema se han propuesto numerosas técnicas que podrían agruparse en dos grandes categorías:

- *Métodos basados en reglas.* Se fundamentan en establecer relaciones entre las diferentes características faciales, como por ejemplo la simetría de la cara [5] [6].
- *Métodos estadísticos.* No asumen ningún tipo de información previa. A partir de un conjunto de muestras de entrenamiento extraen la información relevante que diferencia un objeto cara de uno que no lo es [7] [8]. En este grupo se incluye uno de los métodos más utilizados, el Adaboost [9].

De estos dos grandes grupos, Adaboost ha sido el algoritmo más ampliamente utilizado en el terreno de la detección facial. Viola & Jones presentaron el

método de la detección facial basado en Adaboost [9]. Para ello, presentaban un nuevo conjunto de características que permitía aprender relaciones entre las diferentes partes que componen una cara. El método se puede considerar una extensión de un clasificador genérico al problema de la detección de objetos en imágenes. En su trabajo, Viola & Jones demuestran como a partir de características locales basadas en el cambio de intensidad se podía desarrollar un detector de caras muy robusto. La idea inicial es determinar características basadas en sumas y restas de los niveles de intensidad de la imagen. Para esto se utilizan filtros de Haar de un cierto tamaño y calculados para las posiciones concretas de la sub-imagen que se quiere clasificar. Estas características son evaluadas por un clasificador débil, para decidir si la sub-imagen podría corresponder a una cara o no, tal y como muestra la figura 1. Si dicho valor esta por encima de cierto valor umbral, la ventana se clasificará como cara.

Los clasificadores débiles consiguen unos resultados muy pobres, pero si unimos varios módulos como el de la figura 1, se pueden generar clasificadores más robustos que incrementan exponencialmente el éxito de detección. De esta manera conseguimos clasificadores fuertes, los cuales pueden llegar a una tasa de detección de caras del 99 %. A pesar de ello presentan una desventaja, y es que la tasa de falsas detecciones puede sobrepasar el 30 %.

Por este motivo Viola & Jones propusieron un esquema basado en una cascada de clasificadores fuertes, tal y como muestra la figura 2. Se basa en la concatenación de clasificadores fuertes y está entrenada con todos los ejemplos que la etapa anterior no pudo clasificar correctamente. Por tanto, en cada etapa se entrena un conjunto óptimo de características capaces de detectar cada vez ejemplos más complicados. Básicamente, las primeras etapas se encargan de detectar sub-imágenes que son muy diferentes de una cara, mientras que las últimas, se encargan de rechazar ejemplos mucho más complicados.

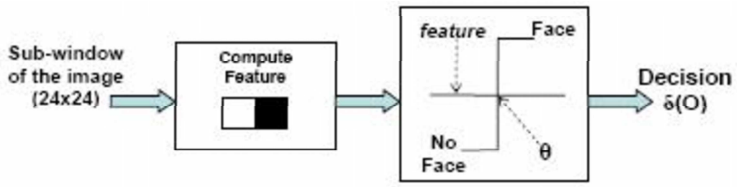


Figura 1. Características Haar-like.

Por otro lado, para conseguir realizar el seguimiento de las caras que se encuentran en las secuencias a analizar, en lugar de usar métodos de *tracking* basados en el movimiento o *flujo óptico*, este proyecto se ha centrado en la detección y seguimiento de puntos faciales de interés. A los puntos de interés también se los denomina puntos salientes, y se definen por tener cierta discriminabilidad visual. En este sentido, los ojos, nariz o boca, definidos por diferentes niveles de gris y contornos que los hacen visualmente salientes, son especialmente indicados para estos procesos. El objetivo de este punto se ha

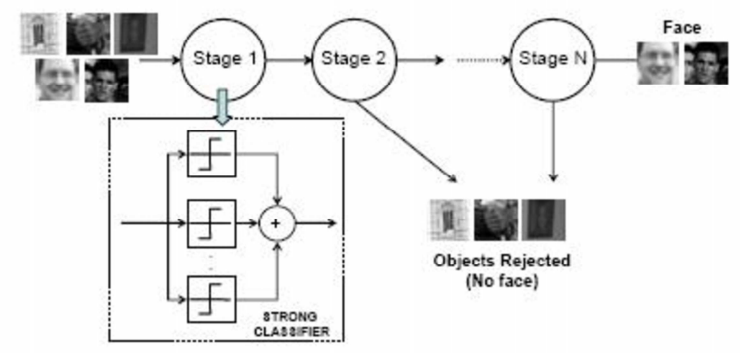


Figura 2. Adaboost: cascada de clasificadores.

centrado en el trabajo Salient Regions [3]. Este método permite detectar regiones salientes de una imagen a partir del nivel de aglomeración de niveles de gris que votan dentro de una región, calculado como la entropía del histograma de grises. Otro de los objetivos en este punto ha sido adaptar las regiones salientes aplicando criterios más robustos que sirvan para complementar su aplicabilidad a la detección de características faciales.

Otro de los grandes problemas en muchas tareas de Visión por Computador es la descripción de regiones de interés. Los descriptores pueden utilizarse para resolver problemas como la detección, clasificación o seguimiento de objetos. Es decir, detectar puntos similares bajo diferentes transformaciones. De esta manera podemos conseguir el seguimiento facial. Una vez se ha detectado la cara, se detectará puntos de interés basándonos en su descriptor, y a medida que dicha cara vaya cambiando con el tiempo, nos centraremos en los nuevos puntos para poder ver como evolucionan tales descripciones.

Podríamos decir que la etapa de detección de los puntos de interés corresponde a la etapa de visión *preatentiva* del Sistema Visual Humano. Esta visión preatentiva o procesamiento visual, preatentivo se realiza automáticamente en la totalidad del campo visual detectando las características básicas de los objetos que vemos. Las características incluyen colores, forma, límites del objeto, contraste, inclinación, tamaño y curvatura, entre otras. Posteriormente estas características son transformadas en objetos coherentes. Este procesamiento se realiza rápidamente y sin esfuerzo por el Sistema Visual Humano.

Las diferentes etapas del HVS han llevado a la creación de diferentes grupos en la comunidad de la visión por computador, los cuales defienden diferentes teorías. Uno de los principales modelos fue atribuido a Neisser en 1964 [19]. Neisser defendía que había dos etapas, la preatentiva y la atentiva. En la primera etapa, la preatentiva, sólo se detectan características (como las comentadas anteriormente). Es decir, se definen regiones locales que producen o presentan alguna discontinuidad espacial. En la etapa atentiva, se encuentran relaciones entre estas características, y se lleva a cabo la agrupación. Este

modelo ha influido ampliamente en la comunidad de la visión por computador (principalmente a través de la labor de Marr en 1982) [14], y es la que como estamos comentando, se sigue en este proyecto. Por lo tanto, se puede decir que los puntos de interés se refiere a la idea que ciertas partes de una imagen o escena son preatentivamente distintivas y, de alguna forma, despiertan la atención visual dentro de las tempranas etapas del Sistema Visual Humano.

El término *salient feature*, se ha usado por muchos otros investigadores, y aunque las definiciones varíen, por intuición, podríamos decir que corresponde a la rareza de un rasgo o punto de interés detectado en las etapas tempranas del HVS. Los detectores de puntos de interés se han usado en múltiples aplicaciones, como en la clasificación de objetos. Uno de los detectores más conocidos es el detector de Harris [12].

Como siguiente punto de este trabajo, se han linkado los diferentes frames analizados. Es decir, una vez realizada la detección de caras y sus características internas, debemos asociarlas con el fin de conseguir su correcto seguimiento temporal. En este sentido, nos hemos centrado en el descriptor SIFT [4]. Este descriptor es el más usado actualmente dada su facilidad de cómputo y alta robustez. El objetivo será asociar las regiones salientes detectadas a partir de minimizar la distancia de sus descriptores SIFT. Este último proceso permite realizar la asociación entre la visión preatentiva y atentiva de nuestro sistema.

Finalmente, como último punto a destacar, se ha realizado una fuente de datos pública, que además de ayudar a procesar los algoritmos de este proyecto, será publicada para divulgar su uso entre los diferentes desarrolladores del área. Los datos son secuencias de vídeo etiquetadas donde diferentes individuos interactúen, de manera que podrán ser usadas tanto para evaluación de detectores faciales y de características como para detección de expresiones, interacciones, género o edad.

El sistema diseñado se ha testeado sobre bases de datos públicas y sintéticas, mostrando gran robustez en la detección y seguimiento de características faciales, simulando el comportamiento preatentivo y atentivo del Sistema Visual Humano y siendo de gran utilidad para propósitos de vídeo vigilancia, interacción humana e interacción hombre-máquina.

Este trabajo se divide en varias partes diferenciadas. Primero se centra en la detección de caras mediante el método del Adaboost. Una vez acabado, se sigue con el algoritmo *Regiones Salientes Complejas*, que se utiliza para la detección de zonas de interés mediante dos características primarias, los niveles de grises y la orientación de los píxels de la imagen. Una vez se detectan las zonas de interés, veremos como se realiza el tracking de éstas, mediante un algoritmo que denominamos *Pseudo-Sift*, basado en el original descriptor SIFT de David Lowe. A partir de aquí, veremos los datos para la creación

de una base de datos pública, con la que tanto nosotros como otros investigadores y proyectistas podrán testear programas de reconocimiento facial y/o de características faciales.

2. Detección de caras

En este apartado nos centraremos en la detección facial, primer paso de este proyecto. Antes de poder hacer el reconocimiento y seguimiento de las características faciales, se detectará la cara o caras existentes en uno de los frames iniciales. De esta manera, la búsqueda de zonas de interés estará más acotada, y así se evita que se detecten zonas de interés fuera de la cara que no nos interesen.

2.1. Detección mediante Adaboost

Para la detección mediante Adaboost se utiliza la variante Gentle Adaboost (figura 6), que es la que ha demostrado recientemente obtener mejor rendimiento en problemas reales [2]. El procedimiento convencional de Adaboost puede ser interpretado fácilmente como el proceso de selección de características greedy (búsqueda exhaustiva secuencial). Se tiene que considerar el problema principal del boosting, en el que un gran conjunto de funciones de clasificación se combinan mediante el voto mayoritario. El objetivo es asociar los pesos grandes con cada función óptima de clasificación y los de menor peso con las funciones que no clasifican correctamente. Adaboost es un mecanismo exhaustivo para obtener la selección de un pequeño conjunto de buenas funciones de clasificación que no sufren variaciones significativas. Por poner una analogía, entre clasificadores débiles y de características, Adaboost es un procedimiento efectivo para buscar un número pequeño de buenas características que no tienen una variación significativa. Un método práctico para completar esta analogía es restringir el clasificador débil para el conjunto de funciones de clasificación, donde cada uno depende de una única característica. Para lograr este objetivo, el algoritmo de aprendizaje débil está diseñado para seleccionar la característica que mejor separa los ejemplos positivos de los negativos [1]. Con tal de calcular estas características rápidamente en varias escalas, se introduce la imagen integral (la imagen integral es muy parecida a la Summed Area Table o SAT, utilizada en gráficos por computador [25] para mapear texturas). La imagen integral puede ser calculada en una imagen empleando unas cuantas operaciones por píxel.

Con el objetivo de obtener un detector robusto en lugar de tratar directamente con los píxels de la imagen, se emplean las características Haar-like. Con este método hay diferencias entre la suma de todos los píxels en algunas regiones rectangulares contiguas de la imagen. Estas características tienen la ventaja de ser invariantes a la iluminación y a la escala, además de ser muy robustas frente al ruido de la imagen.

Hay muchas motivaciones para usar estas características en lugar de los píxels directamente. La razón más común es que estas características pueden actuar para codificar *ad-hoc* el dominio de conocimiento que es difícil de aprender usando una cantidad finita de datos de entrenamiento. Para este sistema hay también una segunda motivación muy importante para decantarse por las características: Los sistemas basados en características operan mucho más rápido que un sistema basado en píxel.

En [27], Lienhart y Maydt extienden el conjunto de características utilizado por Viola y Jones, añadiéndole las versiones rotadas de cada tipo de característica (figura 3). Todas estas características pueden ser calculadas a través de la imagen integral o *SAT1* y la imagen integral rotada 45^o o *RSAT2*. Ambas imágenes auxiliares pueden ser calculadas utilizando únicamente un paso de izquierda a derecha y de arriba a abajo sobre todos los píxels. En la imagen *SAT*, cada píxel *SAT* contiene la suma de todos los píxels del rectángulo de arriba a la derecha, cuyo rango sería desde la esquina de arriba a la izquierda hasta la esquina de abajo a la derecha en (x, y) (figura 4). La imagen *RSAT* se define como la suma de todos los píxels del cuadrado rotado 45^o entre los márgenes de coordenadas mostrados en la figura 4. Dada una ventana de un tamaño fijo a analizar, el conjunto de características será aquel compuesto por todas aquellas características que se puedan generar dentro de la región en un rango de tamaños y posiciones determinado a priori.

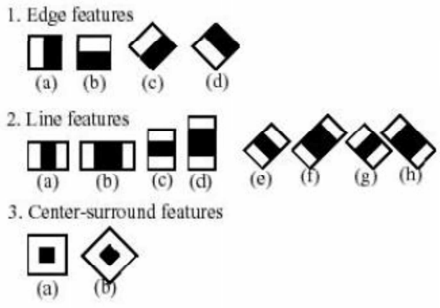
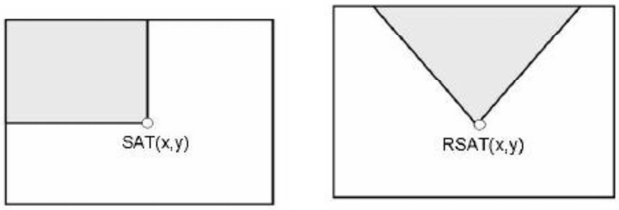


Figura 3. Conjunto extendido de las Haar-like features. La zona blanca corresponde a la región positiva y la zona negra a la región negativa.



(a) (b)

Figura 4. (a) Definición de Summed Area Table (SAT). (b) Definición de Rotated Summed Area Table (RSAT).

Usando la imagen integral, cualquier suma rectangular puede ser calculada

en cuatro arrays de referencias (figura 4). Por ello, la diferencia entre dos sumas rectangulares puede ser calculada en ocho referencias. Dado que los dos rectángulos de características definidos arriba involucran la suma rectangular de adyacentes, pueden ser calculadas en seis arrays de referencias, ocho en el caso de los 3-rectángulos de características, y nueve para los 4-rectángulos de características. Una motivación alternativa para la imagen integral viene del trabajo de Simard [28]. Los autores apuntan que en el caso de operaciones lineales, (p.ej. fog), cualquier operación lineal invertible puede ser aplicada a fog, si su inversa es aplicada al resultado.

Una vez calculada, cualquier característica Haar-like que usemos para el entrenamiento facial, puede ser calculada a cualquier escala o localización en tiempo constante.

En una subventana de una imagen, el número total de características Haar-like es muy grande, mucho más grande que el número de píxels. Con tal de asegurar una clasificación rápida, el proceso de aprendizaje debe excluir la gran mayoría de las características disponibles, y centrarse en un conjunto pequeño de características críticas. La selección de características se consigue a través de una simple modificación en el procedimiento del Adaboost: el aprendiz débil está tan forzado, que cada clasificador débil devuelto puede depender sólo de una única característica. El resultado de cada fase del proceso de boosting, que selecciona un nuevo clasificador débil, puede ser visto como un proceso de selección de características [26].

La velocidad del detector al focalizar la atención en las regiones de la imagen se incrementa combinando sucesivamente más clasificadores complejos en la estructura de la cascada. Con esta aproximación es posible determinar dónde aparecerá un objeto en la imagen. Por ello, el procesamiento más complejo se reserva para estas posibles regiones. La unidad clave de esta técnica es el ratio de falsos negativos del proceso. Debería incluir todos los casos, o casi todos, de instancias de objetos que son seleccionados por el filtro. Estas cascadas se pueden considerar un árbol de decisión degenerado donde en cada fase un detector es entrenado para detectar casi todos los objetos de interés mientras rechaza los que no lo son (figura 5).

Los sistemas de detección deben cumplir fuertes restricciones, tanto en tasa de aciertos como de fallos. Si se entrena un detector simple con estas restricciones, el número de hipótesis que el método de boosting debe combinar para obtener el resultado es enorme. Utilizando la estructura en cascada, la restricción de la tasa de falsos positivos es compartida junto con la tasa de falsos positivos de la cascada de detectores. Lo mismo puede ser aplicado para el porcentaje de aciertos.

Cada fase analiza sólo los objetos aceptados en las fases previas, de modo que

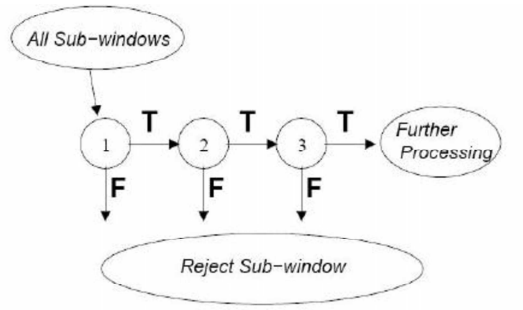


Figura 5. Cascada de clasificadores.

los no-objetos son analizados hasta que son rechazados en una de las fases previas. El número de clasificadores aplicados se reduce exponencialmente debido a la arquitectura de la cascada. Utilizamos Gentle AdaBoost para aprender cada nivel de la cascada y cambiar los objetos rechazados por no-objetos que las fases entrenadas previamente hayan clasificado como correctos (como podemos ver en la figura 6.

Gentle AdaBoost

1. Start with weights $w_i = 1/N$, $i = 1, 2, \ldots, N$, $F(x) = 0$.
2. Repeat for $m = 1, 2, \ldots, M$:
   (a) Fit the regression function $f_m(x)$ by weighted least-squares of y_i to x_i with weights w_i.
   (b) Update $F(x) \leftarrow F(x) + f_m(x)$
   (c) Update $w_i \leftarrow w_i e^{-y_i f_m(x_i)}$ and renormalize.
3. Output the classifier $\text{sign}[F(x)] = \text{sign}[\sum_{m=1}^{M} f_m(x)]$

Figura 6. Algoritmo Gentle Adaboost

Una vez que se han entrenado las cascadas, el resultado de aplicar dicho algoritmo se puede apreciar en la figura 7, donde podemos ver detecciones sobre imágenes de situaciones reales.

En este proyecto, el método de la detección de caras se ha incluido en una librería implementada en *C++*, llamada *facedetect*, que contiene la implementación del detector de caras mediante las librerías OpenCv e IPL .

OpenCv es una librería de procesamiento de imágenes de código abierto desarrollada inicialmente por Intel. Es gratis para uso comercial y de investigación bajo licencia BSD. La librería es multiplataforma, y se puede ejecutar en Mac OS X, Windows y Linux. Se centra principalmente en el procesamiento de imágenes en tiempo real, que se ayudará de las IPL de intel, rutinas optimizadas para acelerar los procesos.

La librería IPL o Image Processing Library es una plataforma independiente

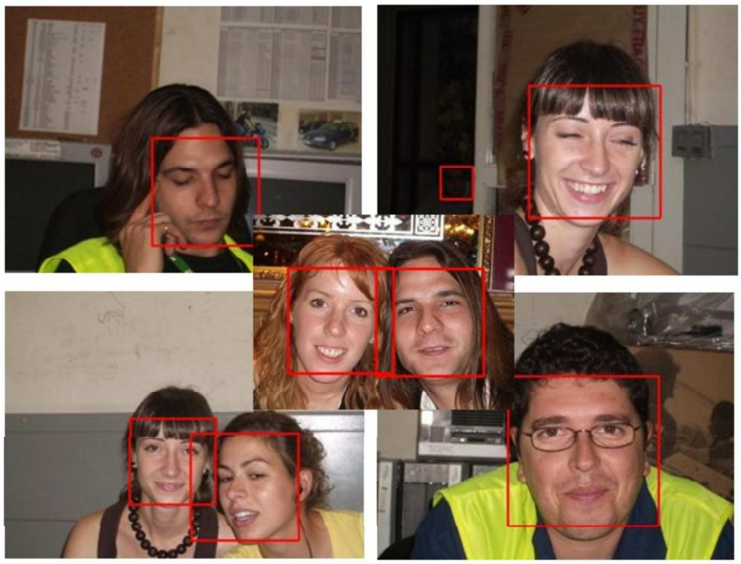


Figura 7. Ejemplos de detecciones en situaciones reales.

de imágenes. El propósito de ésta es ser útil para la combinación del procesamiento de imágenes hecho a medida y la interpretación con métodos estándares para la adquisición, procesamiento, visualización y almacenamiento de la información de la imagen.

La Cascada de clasificación (CvHaarClassifierCascade) utilizada para la implementación del detector de caras ha sido haarcascade *frontalface_alt_tree*. Existen varias cascadas diferentes, pero en las pruebas realizadas sobre diferentes imágenes, haarcascade *frontalface_alt_tree* es la que ha ofrecido mejores resultados.

3. Regiones Salientes Complejas

Como se ha comentado con anterioridad, para conseguir realizar el seguimiento de caras, en lugar de usar métodos de *tracking* basados en movimiento o *flujo óptico*, nos centramos en detectar y hacer el seguimiento de puntos de interés. Estos puntos se definen por tener cierta discriminabilidad visual. En una cara, los puntos indicados para este proceso por sus diferencias en niveles de grises y contornos significativos son por ejemplo los ojos, la nariz o la boca. Para la detección de estos puntos de interés nos hemos basado en el trabajo *Salient Regions* [3]. Este método nos permite detectar regiones salientes de una imagen a partir del nivel de aglomeración de niveles de gris que votan dentro de la región, calculado como la entropía del histograma de grises.

3.1. CSR de niveles de gris

Visual Saliency [15] es un término que se refiere a que ciertas partes de una escena son preatentivamente distintivas, y llaman la atención de alguna forma en las etapas tempranas de la visión humana. Por otro lado, el término *salient feature*, ha sido usado por muchos investigadores [16] [17] [18], y aunque las definiciones varíen, podemos decir que el *saliente* es la rareza de un rasgo. En la actualidad, se ha prestado más atención a los detectores de puntos de interés con sistemas bioinspirados. Uno de los más importantes, en referencia a esta visión pre-atentiva, ha sido atribuido a Neisser [19], modelo que consiste en etapas pre-atentiva y atentiva. En la primera, simplemente se detectan rasgos o características que podrían definir el objeto, y en la segunda se construye el objeto basándose en las características de la primera etapa. Uno de los detectores de puntos de interés más conocidos actualmente, es el detector de Harris [12]. Se basa en buscar los bordes del objeto, que se mantienen en diferentes escalas, para detectar puntos de interés.

Los detectores de puntos interés tienen múltiples aplicaciones, y como veremos se han utilizado en múltiples proyectos:

- Detectar coincidencias en pistas de audio [20].
- Recuperación de imágenes en bases de datos grandes [21].
- Recuperación de objetos en pistas de video [22].
- Clasificación de objetos [23].

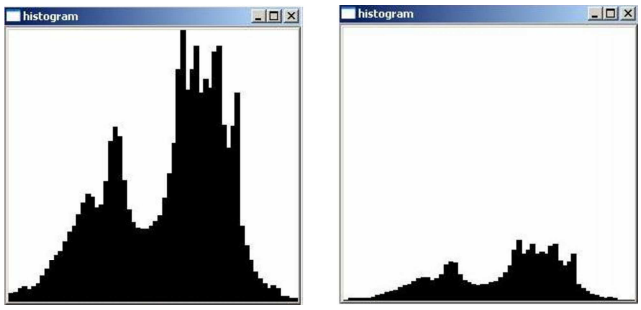
El objetivo de los detectores de puntos de interés es encontrar de un modo no supervisado puntos clave que sean fáciles de extraer, y al mismo tiempo robustos a transformaciones de la imagen. En este caso, nos centraremos en puntos cuya intensidad y estructura local tengan la suficiente singularidad como para decidir si se trata de un punto de interés o no. Para calcular la

complejidad de las regiones hay que basarse en la entropía de los niveles de grises, y una vez calculados, clasificar las regiones por niveles de complejidad. De esta manera, después de testear este método con los métodos actuales existentes, podemos decir que las regiones son altamente discriminativas y menos sensibles a errores que las obtenidas por otros métodos tradicionales.

Inicialmente se recorre la imagen, construyendo para cada píxel un conjunto de regiones a diferentes escalas como se puede ver en la figura 8.

Figura 8. Construcción de las regiones de la imagen.

Una vez se ha construido la región, calculamos su histograma de grises, y realizamos su normalización como se muestra en la figura 9.



(a) (b)

Figura 9. (a) Histograma. (b) Histograma normalizado.

Estos dos procesos se han realizado mediante funciones que ofrecen las librerías de openCV, de manera que no se tendrá que realizar todo el cálculo explicado anteriormente. Lo único que se ha tenido que decidir es el número de bins del histograma. Para ello, se han realizado diferentes pruebas para ver el número *ideal* de bins que optimiza el cálculo de la entropía, paso siguiente de el algoritmo implementado.

Una vez llegados a este punto se calcula la entropía del histograma de grises normalizado. Para calcularla, se utiliza la siguiente ecuación:

$$H(R_x) = -\sum_{i=1}^{n} P_{R_x}(i) log_2 P_{R_x}(i) \tag{1}$$

donde P_{R_X} es la probabilidad que toma la variable d_i en la región local R_x.

El concepto básico de entropía en teoría de la información tiene mucho que ver con la incertidumbre que existe en cualquier experimento o señal aleatoria. Es también la cantidad de *ruido* o *desorden* que contiene o libera un sistema. De esta forma, se podrá hablar de la cantidad de información que lleva una señal.

Como ejemplo, consideremos algún texto escrito en español, codificado como una cadena de letras, espacios y signos de puntuación (la señal será una cadena de caracteres). Ya que, estadísticamente, algunos caracteres no son muy comunes (por ejemplo, y), mientras otros sí lo son (como la a), la cadena de caracteres no será tan *aleatoria* como podría llegar a ser. Obviamente, no se puede predecir con exactitud cuál será el siguiente carácter en la cadena, y eso la haría aparentemente aleatoria. Pero es la entropía la encargada de medir precisamente esa aleatoriedad, y fue presentada por Shannon en su artículo de 1948 A Mathematical Theory of Communication [32].

Shannon ofrece una definición de entropía que satisface las siguientes afirmaciones:

- La medida de información debe ser proporcional (continua). Es decir, un cambio pequeño en una de las probabilidades de aparición de uno de los elementos de la señal debe alterar mínimamente la entropía.
- Si todos los elementos de la señal son equiprobables a la hora de aparecer, entonces la entropía será máxima.

Este proceso hay que repetirlo para cada una de las subregiones que se han construido para dicha imagen. Al acabar con todas ellas, se obtendrán todas sus entropías. El siguiente paso será calcular los máximos para cada uno de los puntos con la siguiente ecuación:

$$W(s,x) = s\frac{|H(s-1,x) - H(s,x)| + |H(s+1,x) - H(s,x)|}{2} \tag{2}$$

donde hay que considerar que para cada escala $s \in S$ y píxel x, se estima W en un caso discreto como una función de cambio en la magnitud de la entropía.

Dichos máximos proporcionarán los puntos de interés de la imagen. Después

de realizar varias pruebas, se decidió que había una posible solución más óptima. Dicha solución consiste en utilizar este método en una área de la imagen reducida. Se quieren encontrar puntos de interés en la caras detectadas en la imagen, por lo tanto, primero se pasará a la imagen un detector de caras y sobre ellas se aplicará el algoritmo *CSR*, y de este modo se mejorará considerablemente el tiempo de cómputo.

Una vez se llegó a este punto se decidió que no interesaban todos los puntos de interés que se conseguían al seguir los pasos anteriores, ya que se detectan muchas zonas que no corresponden a las características que se buscan. Los resultados obtenidos hasta el momento se muestran en la figura 10, que empezaban a ser unos resultados robustos para intentar realizar un seguimiento facial, si se conseguían reducir de alguna manera la cantidad de puntos de interés localizados.

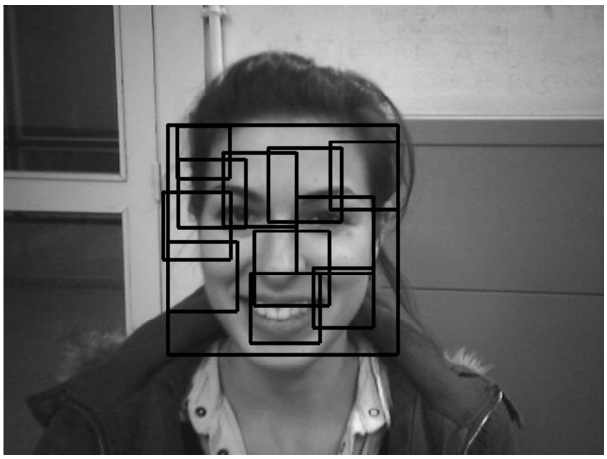


Figura 10. Detección de puntos de interés faciales.

Después de realizar múltiples pruebas regulando los diferentes parámetros de entrada, se llegó a resultados que se creyó que eran realmente positivos como podremos comprobar en el apartado de resultados.

Uno de los valores que se decidió regular fue el tanto por ciento de entropías mayores que se querían visualizar. Para ello lo primero que se implementó, fue un clustering de todos aquellos puntos que estábamos obteniendo. Para realizar este paso se consideraban todos los máximos del espacio de entropía para hacer agrupaciones de regiones basadas en los vecinos más cercanos.

3.1.1. CSR de niveles de gris y CSR de orientaciones

En la sección previa se usa la entropía de niveles de grises para definir la complejidad de saliente de una región dada. Sin embargo, esta aproximación suele fallar en casos donde las regiones tienen complejidades diferentes. En la figura 11 se pueden observar diferentes regiones con la misma cantidad de

píxels para cada nivel de gris y de diferente complejidad visual. Véase que la aproximación basada en la entropía de niveles de grises propuesta por [15] proporciona el mismo valor de entropía, así como el mismo nivel de saliente para todos ellos.

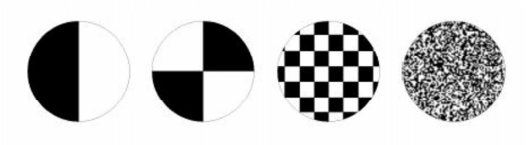
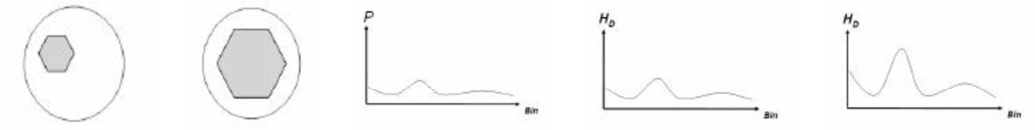


Figura 11. Regiones de diferente complejidad con los mismos niveles de grises.

Una buena medida para solucionar esta patología es la utilización de información complementaria, como podría ser la orientación. Sin embargo, el empleo de orientaciones como una medida de complejidad implica varios problemas. Por ejemplo, supongamos que tenemos las regiones (a) y (b) de la figura 12.



(a) (b) (c) (d) (e)

Figura 12. (a)(b) Dos regiones circulares con el mismo contenido pero resoluciones diferentes. (c) La misma PDF para ambas regiones (a) y (b). (d) Histograma de orientaciones (a). (e) Histograma de orientaciones para (b).

Ambas regiones tienen la misma función de densidad de probabilidades (PDF) (c) de la figura 12, pero tienen diferente número de orientaciones significativas (histogramas de la figura 12 (d) y (e)). En un histograma regular, el gradiente de magnitud bajo es sobre todo debido al ruido, y es distribuido uniformemente sobre todos sus *bins*. Sin embargo, la PDF obtenida de ambas imágenes es la misma debido a la normalización del histograma. Se tuvieron en cuenta estas cuestiones y se incorporó un nuevo procedimiento de normalización de orientaciones nuevo que evalúa correctamente el nivel de complejidad de cada región de la imagen.

3.1.2. Medida normalizada de la entropía de orientaciones

La medida normalizada de la entropía de orientaciones está basada en el cálculo de la entropía usando un pseudo-histograma de orientaciones. El modo habitual de estimar el histograma de orientaciones de una región es usando un rango de 0 a 2π radianes. Considerar la orientación independiente de la magnitud del gradiente evita el riesgo de mezclar la señal con el ruido. En condiciones límite, cuando el gradiente es cero, se obtiene una función de orientación singular. Por otro lado, estos píxels normalmente corresponden a las

regiones homogéneas que pueden ser útiles para describir las partes de los objetos. Para solucionar este problema, se propone introducir un *bin* adicional que corresponde a los píxels con orientación indeterminada, que conoceremos como el *bin* de orientación nula. De este modo, la señal no se mezclará con ruido, y al mismo tiempo, se tendrán en cuenta regiones homogéneas. Por lo tanto se propone calcular el saliente incluyendo la orientaciones nulas en la función de densidad de probabilidades de orientaciones modificada.

Desde el punto de vista práctico, primero de todo se calculan las magnitudes de gradiente relevantes de una imagen para obtener las orientaciones significativas. En vez de usar un umbral experimental, se propone un umbral de orientación adaptativa para cada imagen en particular. Para una imagen dada, dicho método calcula y normaliza el módulo de gradiente $|\nabla(I)|$ en el rango $[0, .., 1]$. Entonces, se estima su histograma, y se aplica el método de Otsu [24] para obtener el umbral adaptativo para las orientaciones. En la figura 13 se muestran las estimaciones de orientaciones significativas obtenidas para dos muestras.

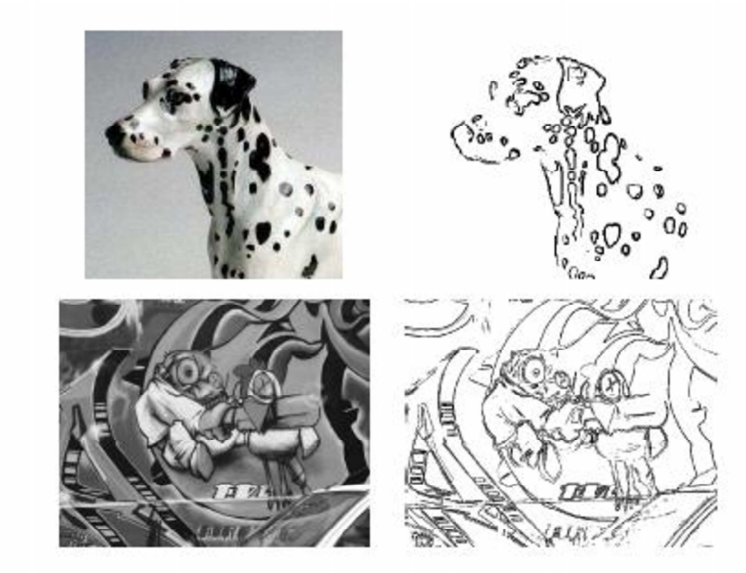


Figura 13. Estimación de orientaciones significativas.

Considerando $k \leq K$, las orientaciones más significativas que usan el umbral adaptativo, donde K es el número total de posiciones en una región dada, se calcula el histograma de orientaciones h_O para n *bins* de orientaciones. En este caso, el número de posiciones de orientaciones nulas se fija en $K - k$, y se añaden al histograma h_O como $h_O(n + 1) = K - k$.

Teniendo en cuenta que la posición $n + 1$ del histograma h_O es el bin de orientaciones nulas, nuestra PDF se obtiene mediante:

$$PDF_O = \frac{h_O(i)}{\sum_{j=1}^{n+1} h_O(j)}, \forall i \in [1, .., n] \tag{3}$$

Finalmente, la función de densidad de probabilidades, PDF_O se utiliza para estimar el valor de entropía de orientaciones de una región dada. Note que el bin $n+1$ o bin de orientaciones nulas no se incluye en la evaluación de la entropía, ya que su objetivo es tan sólo normalizar los primeros n bins del pseudo-histograma de orientaciones.

3.1.3. Combinación del saliente

En este caso particular, el histograma de niveles de grises se combina con el pseudo histograma de orientaciones. Experimentalmente se probó que la combinación de los dos ofrece mejores resultados que si sólo se utiliza el pseudo-histograma de orientaciones o el criterio de la entropía de niveles de grises de forma independiente. De este modo, una vez calculadas las dos PDF correspondientes, se aplican las ecuaciones (1,2 y 3) a cada una de ellas.

El valor final se obtiene mediante la adición simple de $\gamma = \gamma_G + \gamma_O$, donde γ_G y γ_O se calculan mediante la ecuación (1) para los niveles de grises y orientaciones, y γ es el resultado, que contiene las posiciones finales significativas, magnitudes (nivel de complejidad), y las escalas. Otras estrategias, como la del producto o la de las combinaciones logarítmicas de niveles de grises y complejidades de orientación, también han sido testeadas para descubrir zonas de interés. Sin embargo, los resultados no eran satisfactorios ya que estas técnicas fueron pensadas para desechar regiones salientes si uno de los dos valores de saliente es demasiado pequeño, independientemente del dominio del otro componente. Este efecto es insatisfactorio ya que el dominio de un componente sobre el otro puede producir bastante complejidad visual para ser considerada como una región saliente. Por otro lado, una adición simple mostró que se deben mantener las regiones salientes en los casos donde una de las dos medidas es bastante predominante. Al mismo tiempo, esto también permite considerar regiones donde ambos valores de saliente introducen complejidad moderada. El efecto de la medida de saliente combinada se muestra en la figura 14.

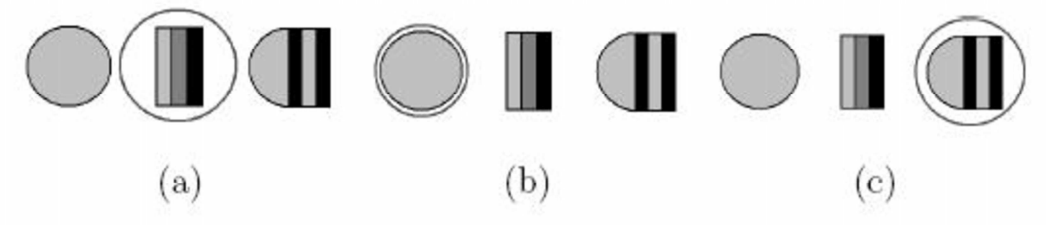


Figura 14. (a) Región de máxima complejidad para entropía de niveles de grises, (b) entropía de orientaciones y (c) entropía combinada.

Esta nueva medida de saliente da un alto valor de complejidad cuando la región contiene diferentes niveles de grises (regiones no homogéneas), y la

complejidad de forma es alta (alto número de magnitudes de gradiente en múltiples orientaciones). La complejidad para estimar el saliente de regiones es la $O(nl)$, donde n es el número de píxels de imagen, y l es el número de escalas buscadas para cada píxel. La complejidad del segundo paso es $O(e)$, donde e es el número de máximos detectados en la etapa anterior. Hay que tener en cuenta que no siempre requieren una búsqueda exhaustiva, y no todos los píxels y escalas posibles tienen que ser verificados o estimados. Sin embargo, la búsqueda exhaustiva es relativamente rápida para calcular (menos de 1 segundo en una imagen de resolución media 800×640).

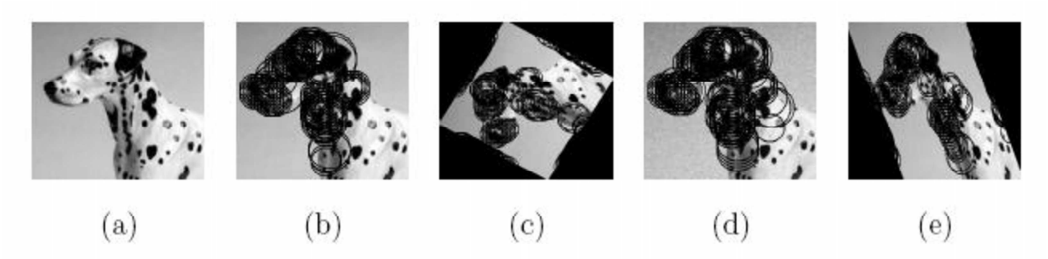


Figura 15. Pruebas del CSR sobre transformaciones sobre imágenes: (a) Imagen original. (b) Detección con CSR, (c) Detecciones sobre imagen rotada. (d) Sobre imagen con ruido. (e) Transformación afín sobre la imagen.

En la figura 15 se muestran ejemplos del CSR para una imagen de muestra sobre la que se aplican diferentes transformaciones.

4. Pseudo-SIFT

En esta sección se presenta la propuesta del algoritmo de descripción de regiones pseudo-SIFT. Para ello, primero se explicará el descriptor SIFT que se ha tomado como base.

4.1. Algoritmo SIFT

El algoritmo SIFT es un método para extraer las características invariantes y distintivas de una imagen que pueden ser usadas para mejorar la correspondencia entre dos vistas diferentes de un objeto o una escena. Las características son invariantes a la escala y rotación de la imagen, y son mostradas para proporcionar un matching robusto entre un rango sustancial de distorsiones afines, cambios en la vista 3D, adición de ruido a la escena y cambio en la iluminación. Las características se distinguen claramente, en el sentido en que una única característica se puede corresponder correctamente con una alta probabilidad contra una gran base de datos de características de muchas imágenes, como se puede ver en la figura 16.

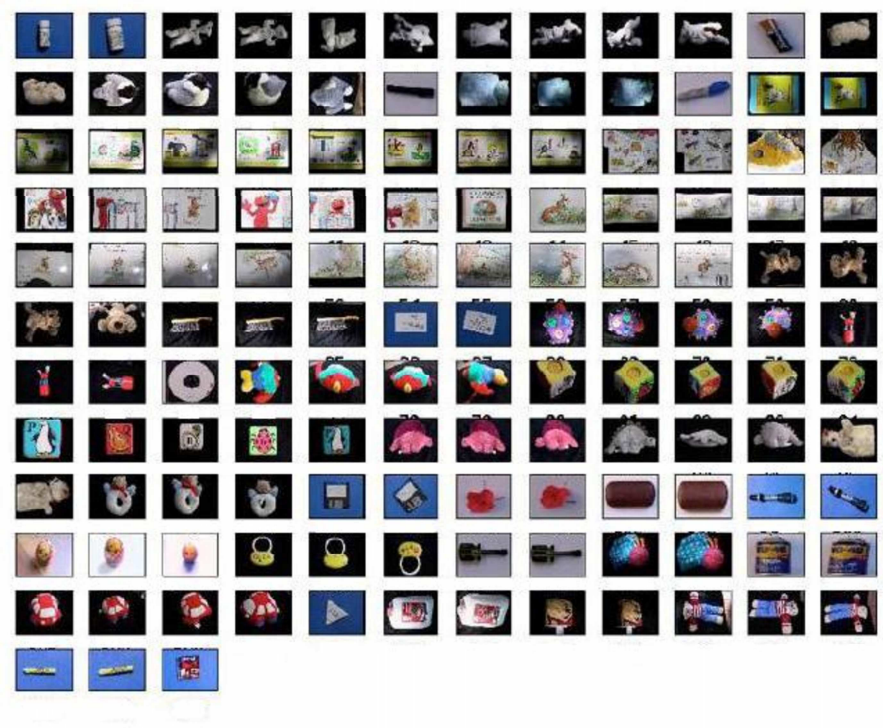


Figura 16. Base de datos de objetos individuales.

El seguimiento consiste en generar el conjunto de características de la imagen:

- Detección del espacio-escala: La primera fase del cálculo busca en todas

las escalas y localizaciones de la imagen. Se implementa eficientemente al emplear la función de diferencia de gaussianas para identificar los puntos de interés que son invariantes a escala y orientación.

- Localización del *keypoint*: en cada localización candidata, un modelo detallado es adecuado para determinar la localización y la escala. Los *keypoints* son seleccionados basándose en su estabilidad.
- Asignación de orientación: Una o más orientaciones se asignan a cada localización de los keypoints basándose en la dirección del gradiente de la imagen local. Todas las futuras operaciones son ejecutadas en los datos de la imagen que ha sido trasformada de acuerdo a la orientación, escala y localización asignadas para cada característica, de este modo se proporciona invariancia a estas transformaciones.
- Descriptor del *keypoint*: Los gradientes de la imagen local son medidos en la escala seleccionada en la región alrededor de cada *keypoint*. Éstos son transformados a una representación que permite, para niveles significativos, distorsión de la forma local y cambios en la iluminación.

Esta aproximación se ha llamado Scale Invariant Feature Transform *(SIFT)*, ya que transforma los datos de la imagen a coordenadas invariantes en la escala relativas a las características locales.

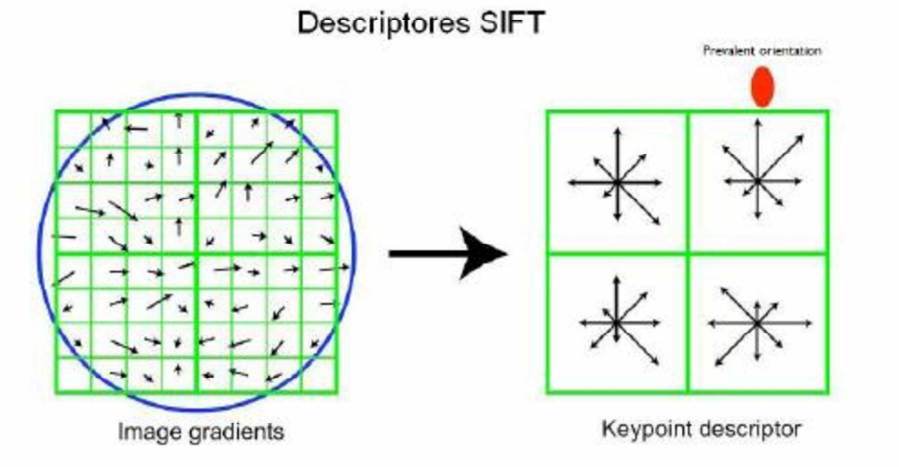


Figura 17. Descriptor SIFT.

En nuestro caso nos hemos basado en la parte del descriptor *SIFT*, que nos servirá para hacer el *matching* de las regiones detectadas en secuencias de vídeo. Una vez tenemos las regiones, el descriptor se crea inicialmente calculando la magnitud del gradiente y la orientación en cada punto de la región con respecto a una región cercana a la localización de dicho punto, tal y como se muestra en la izquierda de la figura 17. Éstos puntos son ponderados por una ventana Gaussiana. Las muestras son acumuladas en un histograma de orientaciones sumando el contenido de 4×4 subregiones, como se ve a la derecha de la figura 17, donde el tamaño de cada flecha corresponde a la suma de las magnitudes de los gradientes cercanos a esa región. La figura muestra

un array descriptor de 2×2 calculado de un conjunto de muestras de 8×8.

4.2. Pseudo-SIFT

El algoritmo pseudo-SIFT está basado en el descriptor SIFT explicado en el apartado anterior, pero con la diferencia de que es una versión adaptativa que en lugar de partir de un número fijo de regiones a dividir la imagen, este número de regiones se adapte dependiendo del dominio de cada problema.

Como se ha explicado en el apartado anterior, en el esquema inicial del SIFT, la diferencia de Gaussianas se usa para detectar los *keypoints*. En nuestro caso, esto no será necesario ya que las regiones de interés se obtienen a través de los detectores de caras y de regiones de interés *(CSR)*, respectivamente.

Una vez se han detectado las zonas de interés, la información que es necesaria extraer dada una imagen es el gradiente y el ángulo. Para ello, primero se calcula para cada píxel las derivadas direccionales en x e y. Opcionalmente se puede aplicar un suavizado a la imagen de tipo gaussiano previo al cálculo de las derivadas.

En el momento en que se obtienen las derivadas direccionales, se calcula el módulo del gradiente para posteriormente normalizarlo entre 0 y 1. Esta normalización se realiza para reducir los efectos de los cambios de iluminación, ya que un cambio en el contraste de la imagen en el que cada valor del píxel es multiplicado por una constante, multiplicará los gradientes por la misma constante, por lo que este cambio de contraste se cancelará gracias al vector de normalización. Un cambio en el brillo en el que una constante es añadida en cada píxel de la imagen no afectará a los valores del gradiente, porque son calculados a partir de las diferencias de píxels. Por lo tanto, el descriptor es invariante a los cambios afines en iluminación. Una vez que se tienen los gradientes G_X y G_Y, se obtiene el ángulo que forman entre ambos.

Para el cálculo del vector descriptor, primero dividiremos la imagen en un patrón de $n \times n$ regiones. En el caso de la imagen 18, por ejemplo, la división será de 4×4 regiones, que numeraremos de 1 a 16.

Como se ha comentado anteriormente, los gradientes son normalizados para no sufrir variación de iluminación afín. Estos cambios pueden causar un gran cambio en magnitudes relativas para algunos gradientes, pero probablemente afectan menos a las orientaciones de los gradientes. Por ello, se reduce la influencia de una magnitud de gradiente grande a través del *threshold* de valor 0,2, y posteriormente renormalizando el vector. Esto significa que la correspondencia de magnitudes para gradientes grandes no es tan importante, y que la distribución de las orientaciones tiene mayor énfasis. El valor de 0,2

Figura 18. División en regiones de la imagen patrón.

fue determinado experimentalmente en [31] usando imágenes con diferencias de iluminaciones para los mismos objetos 3D.

Como ya se ha calculado el módulo y el ángulo previamente de cada píxel, si este módulo supera o iguala al *threshold*, que en nuestro caso será 0,2, se tendrá en cuenta para obtener el descriptor pseudo-SIFT. El ángulo se emplea para conocer el bin al que pertenece el píxel. Se debe actualizar en la región y sus vecinas. La actualización consiste en sumar las distancias euclídeas de la posición del píxel actual y la del centroide de cada una de las regiones (la región a la que pertenece el píxel y las regiones vecinas). Una vez que se tiene este sumatorio, los vectores que corresponden a estas regiones se modifican en bin correspondiente y sus siguientes, sumando el valor anterior y dicho sumatorio.

Una vez que se han recorrido todos los píxels de la imagen y se han realizado todos los cálculos, los vectores que representan a las regiones se unen en un único vector, dando lugar, en el caso de 128 posiciones (16 vectores $\times$ 8 bins/vector = 128 posiciones). Este vector resultante será el descriptor pseudo-SIFT.

El pseudocódigo para la obtención del descriptor pseudo-SIFT sería el siguiente:

- **Paso 1.** *Se calculan los píxels que irán por cada región dependiendo del ancho y el alto de la imagen.*
- **Paso 2.** *Se calcula la región en la que se encuentra el píxel.*
- **Paso 3.** *Se calcula el punto medio x e y por región.*
- **Paso 4.** *Se calculan las regiones vecinas.*
- **Paso 5.** *Mientras no sea el final de la imagen:*
  - **Paso 5.1.** *Si el módulo es mayor que el threshold dado por parámetro.*
  - **Paso 5.2.** *Se calcula en qué bin está el ángulo del píxel y el bin siguiente.*
  - **Paso 5.3.** *Se calculan las distancias Euclídeas del píxel y los centroides de la región a la que pertenece el píxel y también de las regiones vecinas.*
  - **Paso 5.4.** *Se suman todas las distancias.*
  - **Paso 5.5.** *En la región a la que pertenece el píxel y sus vecinas se actual-*

izan únicamente los campos pertenecientes al bin y su siguiente sumando en cada una el valor que ya poseían junto con la suma de las distancias.

- **Paso 5.6.** *Se copia el valor de los vectores creados para las regiones en un único vector que dará lugar al descriptor pseudo-SIFT.*

5. Sistema

En este apartado se verá como se han integrado todos los métodos implementados y explicados con anterioridad. Además veremos los diagramas de ejecución y de clases de los métodos.

Como se puede ver en la figura 19, el proceso del sistema se divide en varios pasos que analizaremos seguidamente en el pseudocódigo.

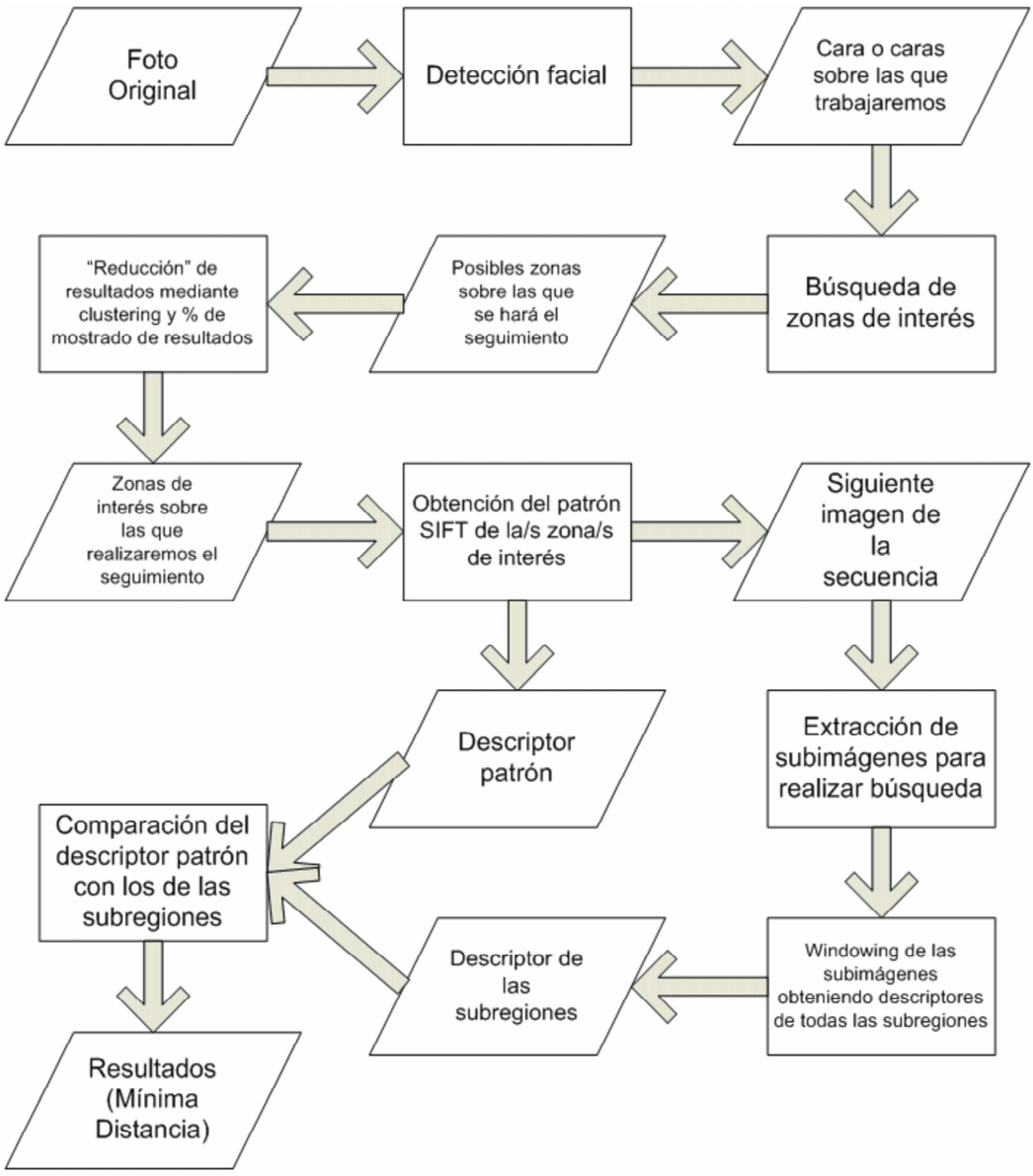


Figura 19. Diagrama de ejecución del sistema.

El pseudocódigo del algoritmo de seguimiento con los pasos más importantes para obtener el seguimiento sobre la secuencia de imágenes se muestra a con-

tinuación :

- **Paso 1.** *Detección de la cara.*
- **Paso 2.** *Detección de zonas de interés (obtención de coordenadas iniciales).*
- **Paso 3.** *Cálculo del descriptor pseudo-SIFT de cada una de las zonas de interés o características faciales.*
- **Paso 4.** *Mientras haya imágenes en la secuencia.*
  - **Paso 4.1.** *Extraer subimagen x veces más grande que la del patrón.*
  - **Paso 4.2.** *Recorrer esta subimagen formando regiones para obtener el descriptor pseudo-SIFT de cada una.*
  - **Paso 4.3.** *Comparar cada descriptor pseudo-SIFT con el del patrón.*
  - **Paso 4.4.** *Remarcar la región que ha obtenido mejor resultado (Característica facial detectada).*
  - **Paso 4.5.** *Actualizar descriptor pseudo-SIFT patrón para comparar con la siguiente imagen.*

En la figura 20 se muestra un ejemplo completo de ejecución del sistema.

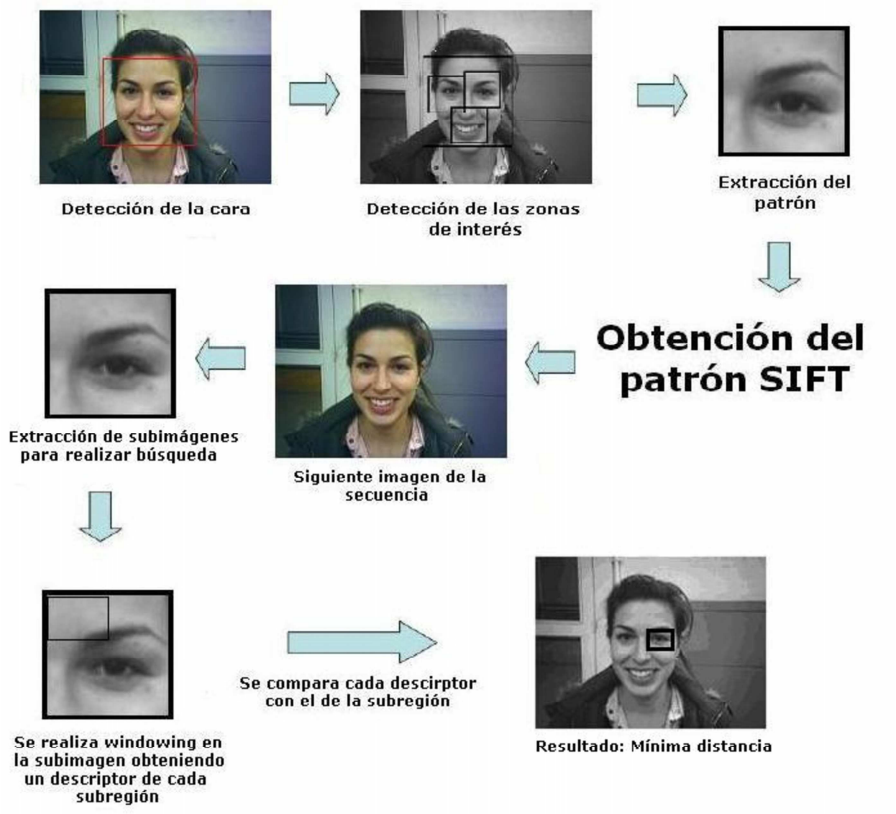


Figura 20. Funcionamiento del sistema paso a paso.

En resumen, dicho algoritmo empieza detectando la cara en la imagen inicial de la secuencia mediante el método *facedetect* (como ya se ha comentado con anterioridad). Con este proceso se consigue delimitar la zona de búsqueda de

zonas de interés, ya que de lo contrario, zonas del entorno podrían detectarse como posibles zonas y este efecto no nos interesa.

Una vez detectada la cara , se pasa a detectar las zonas de interés mediante el algoritmo CSR. Como ya habíamos comentado, se hace una primera detección gracias a los cambios de niveles de grises. Seguidamente se añaden las orientaciones, y con este proceso, después de realizar el clustering y mostrar un porcentaje de los puntos obtenidos, conseguimos que queden las zonas que realmente nos interesan. A continuación, se procede a calcular el descriptor *pseudo-SIFT* del patrón de cada zona de interés como podemos ver en la figura 21.

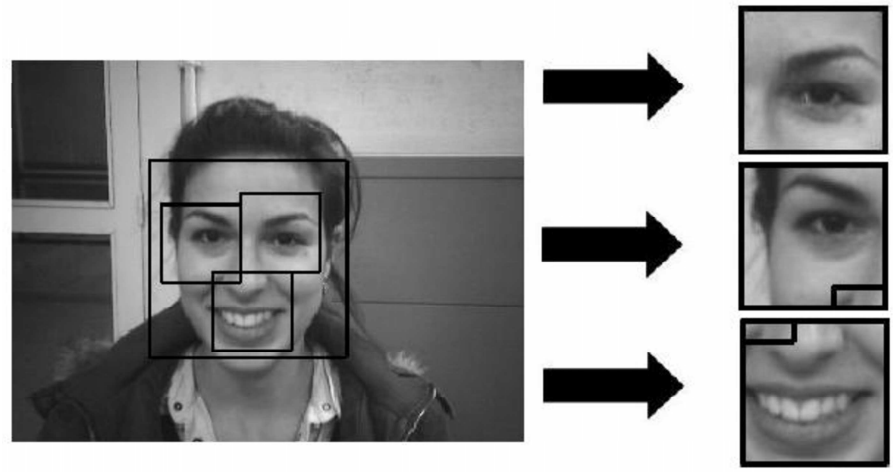


Figura 21. Extracción de los patrones a partir de la imagen inicial.

Todo este proceso realiza la descripción de una imagen patrón a partir del *pseudo-SIFT*. Sin embargo, no existe una gran diferencia para obtener el descriptor en las imágenes de una secuencia. Gracias a la detección de de las zonas de interés, es posible obtener las coordenadas en las que se encuentra dicha zona. Para obtener mayor precisión en el seguimiento de la característica, la búsqueda en la imagen siguiente se efectúa en una subregión de la imagen de una proporción indicada por parámetro (figura 22).

Figura 22. Subimagen de búsqueda extraída dependiendo de la proporción.

Esta subimagen se va dividiendo a su vez en subregiones como podemos ver

en la figura 23, y de cada una de ellas se calcula el descriptor pseudo-SIFT, que para la primera imagen se compara con el descriptor obtenido por el patrón, y para las siguientes se comparará con el mejor descriptor pseudo-SIFT más cercano obtenido en las regiones CSR obtenidas en imágenes anteriores y posteriores.

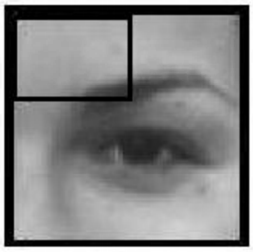


Figura 23. El *windowing* se efectúa en toda la subimagen.

Al extraer el descriptor pseudo-SIFT, también hemos podido adaptar el número óptimo de regiones que nos aporta una mejor descripción de la imagen para cada problema en particular.

Una vez comentada la integración de los diferentes módulos del sistema, en la figura 24 se muestra el diagrama de clases del sistema completo.

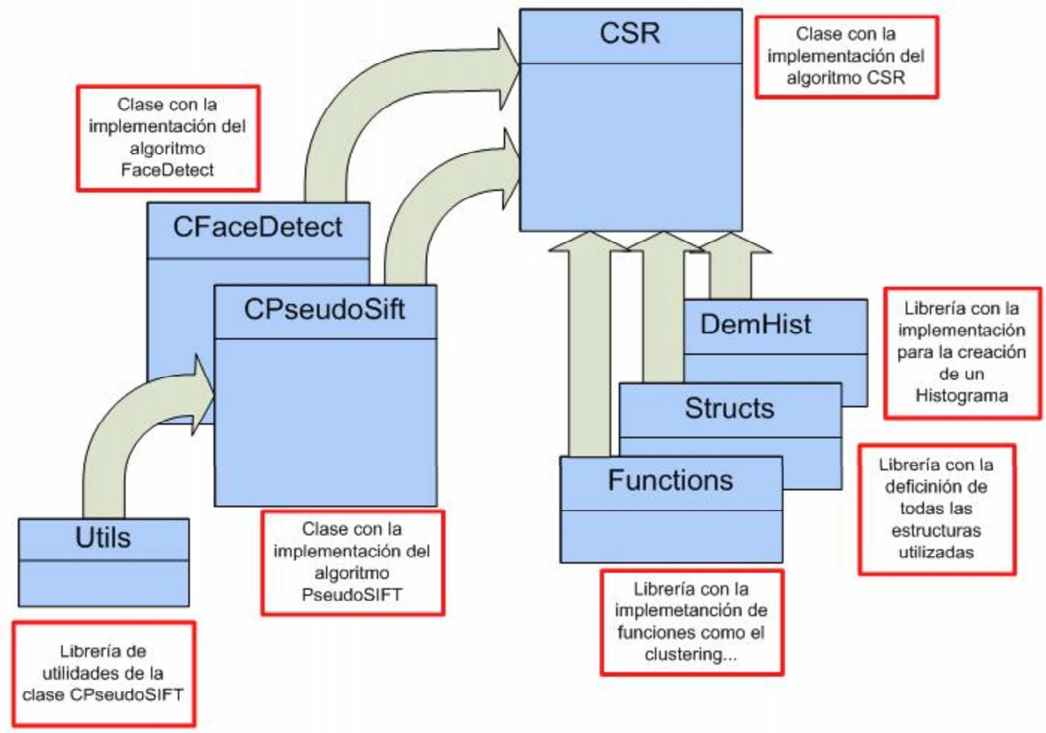


Figura 24. Diagrama de Clases de la Aplicación.

6. Resultados

En este apartado, presentaremos los datos que se han utilizado para realizar las pruebas, los métodos usados en las comparativas y los experimentos realizados para verificar el correcto funcionamiento de la aplicación.

6.1. Datos

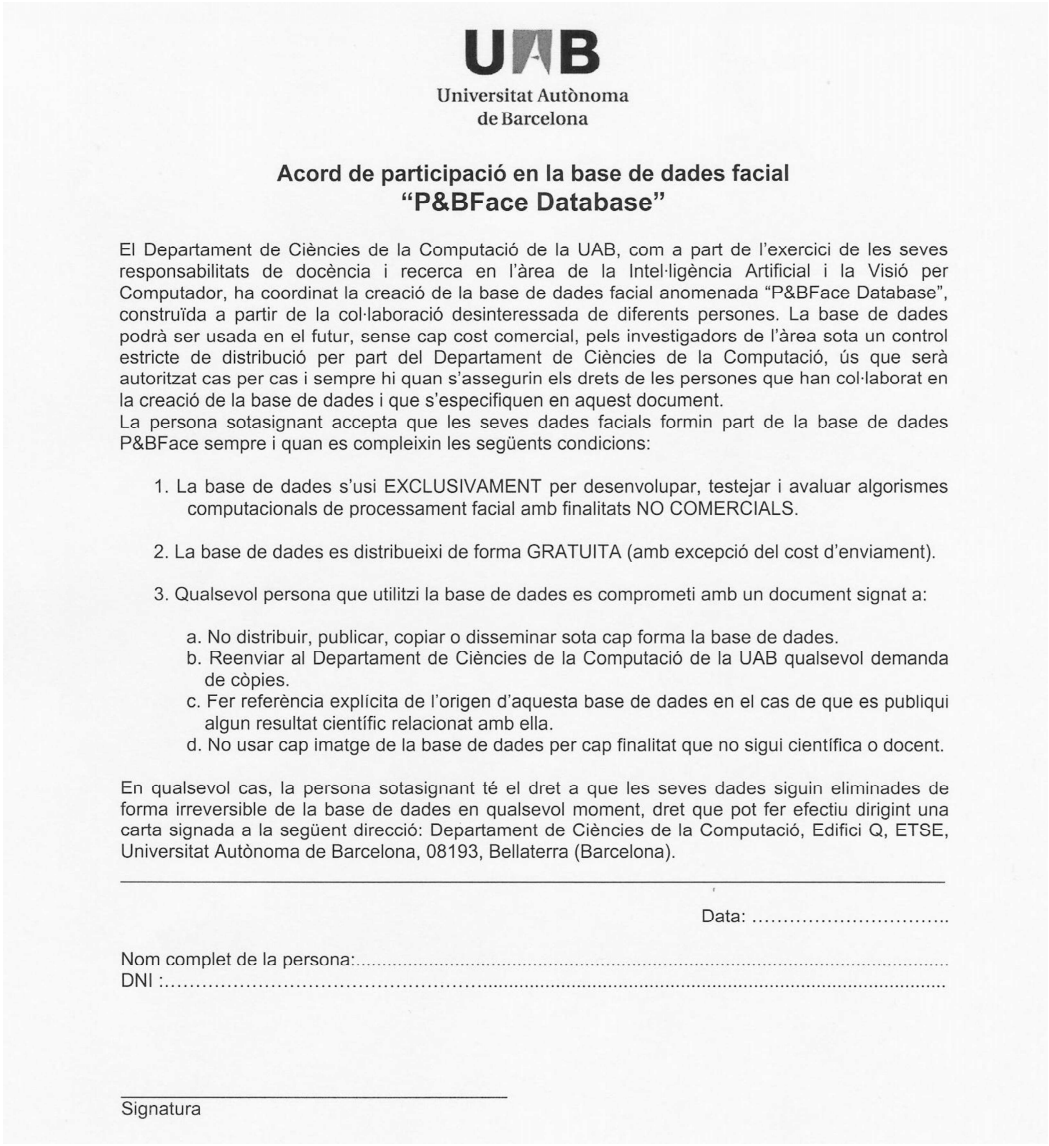
Todos los datos con los que se ha trabajado son fotos, la mayoría de las cuales se hicieron con una cámara digital de 2 Megapíxels, a pesar de que se hicieron pruebas con otros tipos de cámaras para realizar pruebas a distintas resoluciones. Para los experimentos de la detección facial, las fotos utilizadas fueron por un lado de creación propia y por otro se obtuvieron de la base de datos pública face database de la Caltech Repository Database [36].

También se han extraído imágenes reales de voluntarios con una web cam Creative WebCam Notebok Ultra. Dicha cámara permite capturar imágenes y vídeos con facilidad con el dispositivo Creative WebCam o PC-CAM. Con WebCam Center, se puede realizar capturas básicas de imágenes estáticas y vídeo y también realizar tareas avanzadas como supervisión remota, detección de movimiento y captura de vídeo por lapso de tiempo.

Las imágenes se han grabado con una resolución de 352×288 píxels y una velocidad de frame de 30 frames/segundo. Para poder utilizar estas imágenes ha sido necesario que la persona que aparece en ellas haya firmado una autorización, como se puede apreciar en la figura 25. Para cada una de las personas que han participado en estas filmaciones, se han filmado 2 secuencias de vídeo diferentes, variando tanto el lugar como la iluminación, con la intención de obtener la mayor cantidad posible de escenarios y situaciones posibles. Cada individuo, tal y como se observa en la figura 26, debe realizar una secuencia que consiste en:

Dadas cuatro expresiones faciales: neutral, sonrisa, enfado y sorpresa; y las posiciones: frontal, mirar arriba, mirar abajo, mirar a la derecha y mirar a la izquierda; para cada una de las cuatro expresiones E = {neutral, sonrisa, enfado y sorpresa}, el vídeo sigue esta secuencia:

- Posición frontal, hacer expresión E.
- Volver a la expresión neutra.
- Mirar arriba, hacer expresión E.
- Volver a la posición frontal y hacer expresión neutra.
- Mirar abajo, hacer expresión E.
- Volver a la posición frontal y hacer expresión neutra.



UAB
Universitat Autònoma
de Barcelona

Acord de participació en la base de dades facial
"P&BFace Database"

El Departament de Ciències de la Computació de la UAB, com a part de l'exercici de les seves responsabilitats de docència i recerca en l'àrea de la Intel·ligència Artificial i la Visió per Computador, ha coordinat la creació de la base de dades facial anomenada "P&BFace Database", construïda a partir de la col·laboració desinteressada de diferents persones. La base de dades podrà ser usada en el futur, sense cap cost comercial, pels investigadors de l'àrea sota un control estricte de distribució per part del Departament de Ciències de la Computació, ús que serà autoritzat cas per cas i sempre hi quan s'assegurin els drets de les persones que han col·laborat en la creació de la base de dades i que s'especifiquen en aquest document.
La persona sotasignant accepta que les seves dades facials formin part de la base de dades P&BFace sempre i quan es compleixin les següents condicions:

1. La base de dades s'usi EXCLUSIVAMENT per desenvolupar, testejar i avaluar algorismes computacionals de processament facial amb finalitats NO COMERCIALS.

2. La base de dades es distribueixi de forma GRATUITA (amb excepció del cost d'enviament).

3. Qualsevol persona que utilitzi la base de dades es comprometi amb un document signat a:

   a. No distribuir, publicar, copiar o disseminar sota cap forma la base de dades.
   b. Reenviar al Departament de Ciències de la Computació de la UAB qualsevol demanda de còpies.
   c. Fer referència explícita de l'origen d'aquesta base de dades en el cas de que es publiqui algun resultat científic relacionat amb ella.
   d. No usar cap imatge de la base de dades per cap finalitat que no sigui científica o docent.

En qualsevol cas, la persona sotasignant té el dret a que les seves dades siguin eliminades de forma irreversible de la base de dades en qualsevol moment, dret que pot fer efectiu dirigint una carta signada a la següent direcció: Departament de Ciències de la Computació, Edifici Q, ETSE, Universitat Autònoma de Barcelona, 08193, Bellaterra (Barcelona).

Data: .....

Nom complet de la persona:.....
DNI :.....

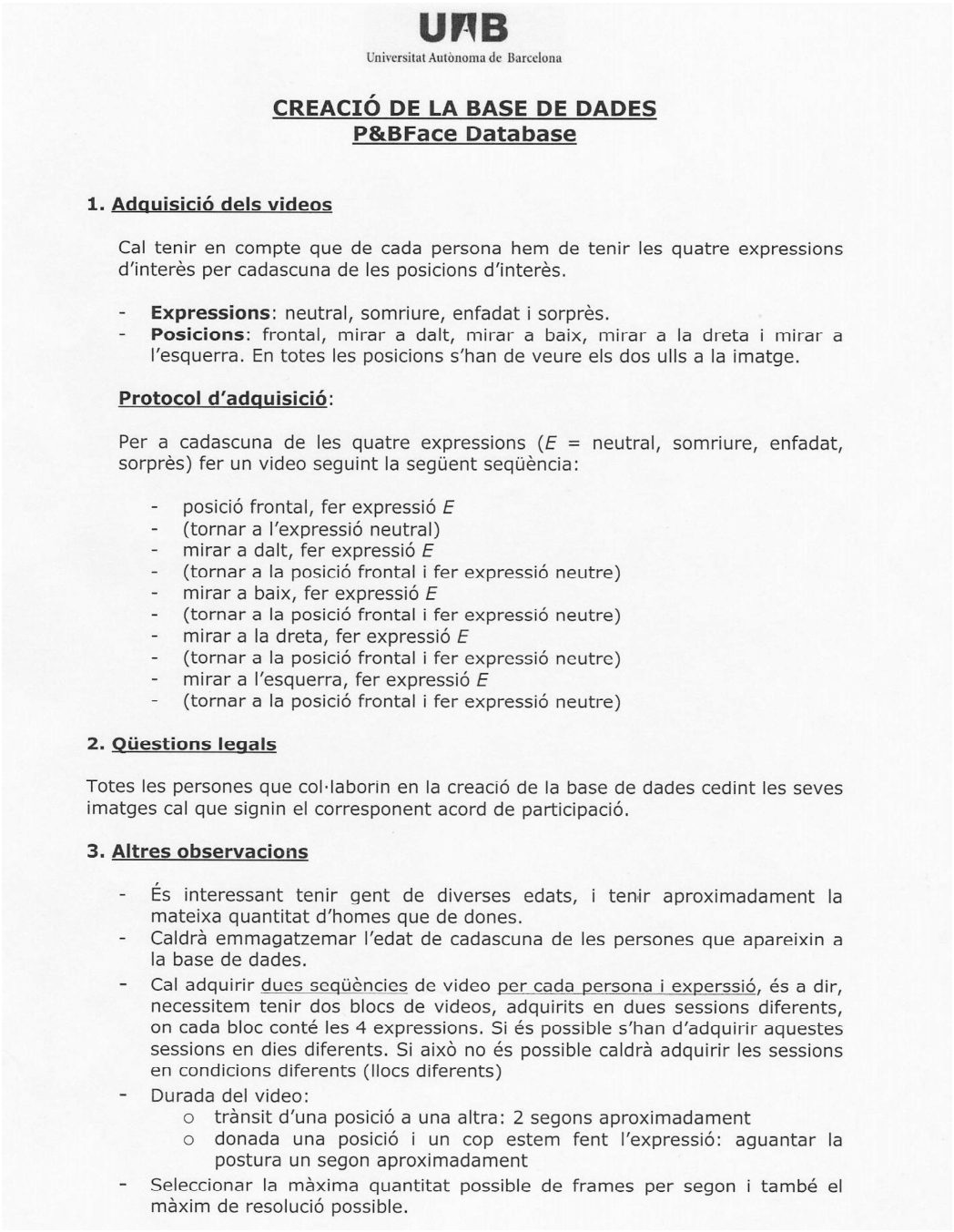
Signatura

Figura 25. Autorización que deben firmar todos los voluntarios para la utilización de su imagen.

- Mirar a la derecha, hacer expresión E.
- Volver a la posición frontal y hacer expresión neutra.
- Mirar a la izquierda, hacer expresión E.
- Volver a la posición frontal y hacer expresión neutra.

Como ejemplos de la realización de esta base de datos podemos ver frames sueltos de diferentes videos como podemos ver en la figura 27 y de una secuencia completa de un video como vemos en la figura 28.

Con estos datos, el Centro de Visión por Computador está implementando una base de datos de vídeos que será utilizada por la comunidad científica para sus investigaciones. En dicha base de datos estarán incluidos los vídeos



UAB
Universitat Autònoma de Barcelona

CREACIÓ DE LA BASE DE DADES
P&BFace Database

1. Adquisició dels videos

Cal tenir en compte que de cada persona hem de tenir les quatre expressions d'interès per cadascuna de les posicions d'interès.

- **Expressions**: neutral, somriure, enfadat i sorprès.
- **Posicions**: frontal, mirar a dalt, mirar a baix, mirar a la dreta i mirar a l'esquerra. En totes les posicions s'han de veure els dos ulls a la imatge.

Protocol d'adquisició:

Per a cadascuna de les quatre expressions (*E* = neutral, somriure, enfadat, sorprès) fer un video seguint la següent seqüència:

- posició frontal, fer expressió *E*
- (tornar a l'expressió neutral)
- mirar a dalt, fer expressió *E*
- (tornar a la posició frontal i fer expressió neutre)
- mirar a baix, fer expressió *E*
- (tornar a la posició frontal i fer expressió neutre)
- mirar a la dreta, fer expressió *E*
- (tornar a la posició frontal i fer expressió neutre)
- mirar a l'esquerra, fer expressió *E*
- (tornar a la posició frontal i fer expressió neutre)

2. Qüestions legals

Totes les persones que col·laborin en la creació de la base de dades cedint les seves imatges cal que signin el corresponent acord de participació.

3. Altres observacions

- És interessant tenir gent de diverses edats, i tenir aproximadament la mateixa quantitat d'homes que de dones.
- Caldrà emmagatzemar l'edat de cadascuna de les persones que apareixin a la base de dades.
- Cal adquirir dues seqüències de video per cada persona i expressió, és a dir, necessitem tenir dos blocs de videos, adquirits en dues sessions diferents, on cada bloc conté les 4 expressions. Si és possible s'han d'adquirir aquestes sessions en dies diferents. Si això no és possible caldrà adquirir les sessions en condicions diferents (llocs diferents)
- Durada del video:
  - trànsit d'una posició a una altra: 2 segons aproximadament
  - donada una posició i un cop estem fent l'expressió: aguantar la postura un segon aproximadament
- Seleccionar la màxima quantitat possible de frames per segon i també el màxim de resolució possible.

Figura 26. Instrucciones para la grabación de los vídeos y creación de la base de datos.

previamente mencionados y también se adjunta por cada uno un archivo XML en el que se incluye la expresión, posición y localización de la característica. De este modo, se tendrán etiquetados los vídeos con todos los datos de interés de los mismos. En la figura 29 se muestra el esquema que se ha seguido para etiquetar todas las características de cada secuencia de vídeo. Actualmente se dispone de 4 secuencias de vídeo de 20 individuos diferentes.

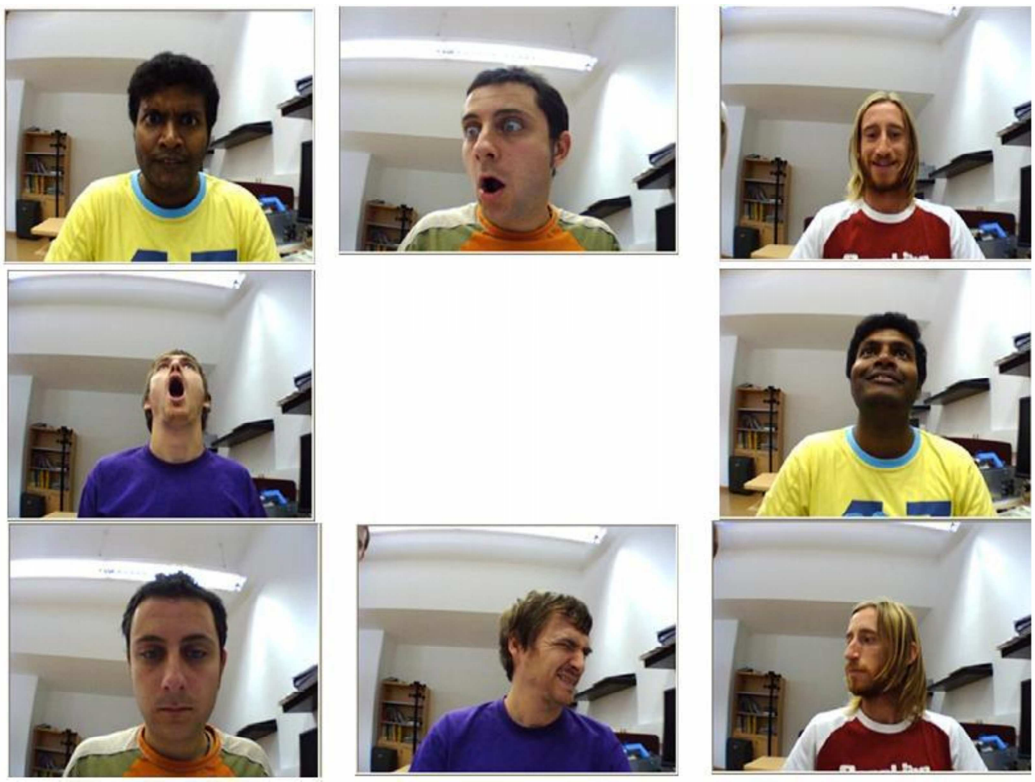


Figura 27. Frames de ejemplo de ORMG Data Set.

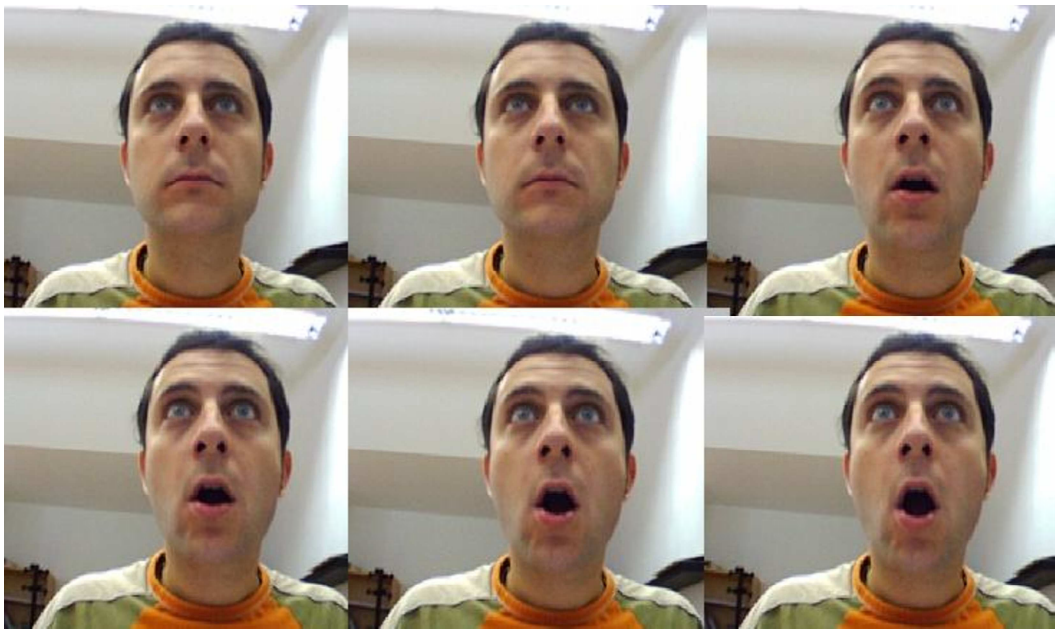


Figura 28. Secuencia de ejemplo de ORMG Data Set.

6.2. Métodos

Todos los métodos utilizados (Facedetect, CSR, SIFT) ya se han explicado con detalle en los apartados anteriores, de manera que en este apartado nos

```
<html>
<head>
<title>Video Labelling</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000">
<face>
   <feature>
        <type> </type>
        <xposition> </xposition>
        <yposition> </yposition>
        <width> </width>
        <heigth> </heigth>
        <state> </state>
   </feature>
</face>


</body>
</html>
```

Figura 29. Esquema del archivo XML de etiquetado de los vídeos.

centraremos en ver las diferentes variables que se pueden regular para llegar a valorar que efectos producen sobre los resultados.

Para la detección facial, como ya hemos visto, se recorre la imagen para poder ir construyendo diferentes regiones y detectar todas las caras que pueda haber en la foto en cuestión. Para ello, podemos definir.

En el caso del FaceDetect:

- La cascada que utilizaremos. En nuestro caso después de realizar pruebas con todas las cascadas disponibles hasta el momento, se decidió usar *haarcascade_frontalface_alt* ya que ha sido la que mejor resultados nos ofrecía, al ser la que menos fallos cometía, a pesar de su tasa de no detecciones (19 aciertos, 2 fallos y 17 no detecciones).

Y en el del CSR:

- El número de píxels que nos desplazamos (horizontal y verticalmente) entre la construcción de dos regiones consecutivas. Según las pruebas realizadas, el valor con el que mejor se comporta el algoritmo es con saltos de 3 píxels en 3 píxels, tanto horizontal como verticalmente.
- El número máximo y mínimo de píxels que ocupará la región. En este caso, los valores escogidos dado al buen funcionamiento del algoritmo son regiones de 20 a 60 píxels.
- El número de píxels incrementados entre el mínimo y máximo de la región. El incremento fijado por óptimos resultados fue 3.
- El *threshold* para unificar los datos en el clustering. Mientras la diferencia

entre máximos vecinos no sea mayor de 40, fusionaremos dichos máximos en uno solo, para trabajar más rápidamente al no tener tantos datos a procesar.
- El porcentaje de zonas de interés a la hora de mostrarlos por pantalla. Después de diferentes pruebas realizadas, se decidió fijar este valor en un 40 %.

Los parámetros que se han fijado para la ejecución del pseudo-SIFT:

- Threshold: Su valor es de 0,2 y se utiliza como umbral para el valor del módulo del gradiente de cada píxel. Si se supera dicho valor, el píxel se tendrá en cuenta para el cálculo del descriptor pseudo-SIFT.
- N: Se corresponde con el número de regiones en que se va a dividir la imagen para el cálculo del descriptor pseudo-SIFT. Está fijado a 4 en horizontal y 4 en vertical, por lo que obtenemos 16 regiones.
- Bins: Es el número de orientaciones en que se divide el array de las regiones. Se ha fijado a 8 después de un análisis, por lo que si dividimos 360 entre 8, tenemos que cada bin abarca 45^o de orientaciones.
- Gauss: Este parámetro es necesario para aplicarle un suavizado o Smooth a la imagen antes del cálculo del gradiente y el ángulo. Se ha determinado valor 3,0.

6.3. Experimentos

En los experimentos previos se ha mostrado la viabilidad de los métodos implementados para realizar la detección facial y de características faciales así como su seguimiento en secuencias de vídeo procedentes de entornos no controlados. Algunos puntos sobre este sistema necesitan ser comentados.

Las pruebas iniciales se realizaron en Matlab con la finalidad de ver su viabilidad de los algoritmos para resolver cada uno de los problemas. Una vez que se vio la viabilidad, estos algoritmos se implementaron en Visual C++ utilizando las librerías adecuadas. Este paso nos permitió optimizar el código para poder procesar las imágenes con más agilidad. No obstante, cuando la mayoría de los procesos necesitan ser ejecutados de forma consecutiva sobre imágenes, el proceso tarda unos 6 segundos sobre una máquina INTEL Pentium M 1.7 Ghz con 1G de RAM.

Decir que poco a poco se fueron haciendo pruebas en puntos medios como la detección facial. Llegados a este punto se realizaron pruebas con todas las cascadas de manera que podíamos ir comprobando cual era la que tendría un funcionamiento óptimo para llevar a cabo este proyecto. Algunos de los experimentos que realizamos en este punto podemos verlos en la figura 30. Y por ejemplo, poder descubrir que con individuos con características como la barba era donde podíamos tener más problemas.

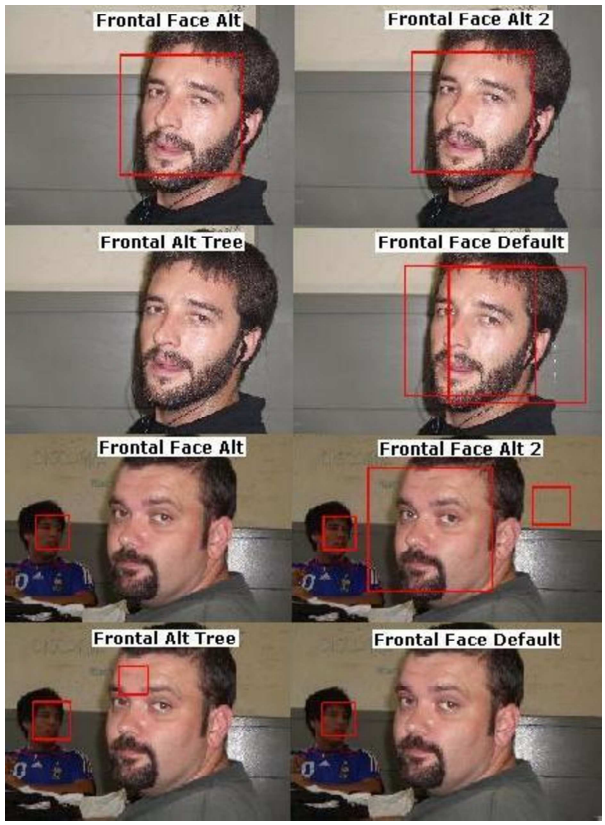


Figura 30. Experimentos de detección facial con diferentes cascadas.

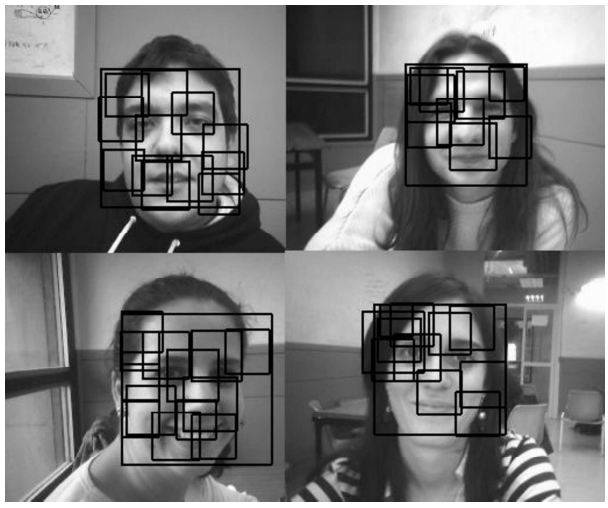


Figura 31. Experimentos CSR niveles de grises.

Una vez solucionado este problema, pasamos a implementar el *CSR*, que se dividió en dos fases. La primera de ellas fue la implementación y el testeo de dicho algoritmo pero limitándolo a la parte de niveles de grises. En esta fase, y después de regular los parámetros explicados anteriormente empezábamos a obtener resultados similares a los de la figura 31.

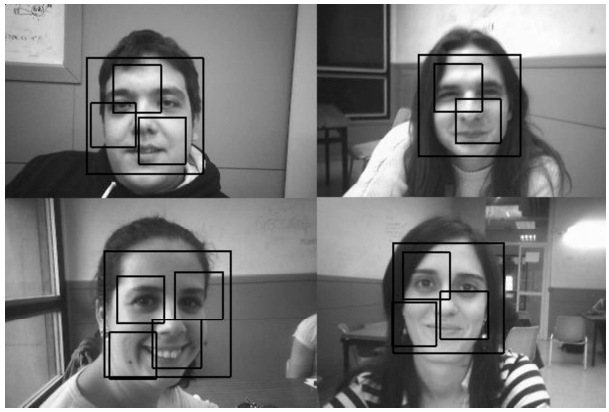


Figura 32. Experimentos CSR niveles de grises y orientaciones.

El siguiente paso fue introducir la parte de orientaciones. Como se había comentado los resultados mejoraban de forma considerable. Empezamos a realizar pruebas con parámetros similares a los del experimento anterior, obteniendo los resultados que se observan en la figura 32.

Dichas pruebas se realizaban sobre imágenes sueltas. Una vez se llegó al punto final del proyecto, y se testeó la parte del *Pseudo-SIFT*, realizando varias pruebas en frames aleatorios, como podemos ver en la figura33. Seguidamente se analizaron los experimentos haciendo la detección sobre el primer frame, y realizando asociaciones con el *Pseudo-SIFT* sobre frames consecutivos. Este proceso podemos verlo en la imagen 34.

Como trabajo pendiente dejamos abierta la optimización de código para implementar estos sistemas en entornos para su funcionamiento en tiempo real. Cabe destacar que además de la optimización de código, el hardware utilizado nos podrá ayudar en esta tarea.

De igual forma, el modelo entero del sistema se deja abierto para ser extendido. En este proyecto las características internas detectadas han sido los ojos, nariz y boca principalmente, datos que nos aportan información acerca del posicionamiento de la persona presente en la imagen. De igual forma, la misma metodología usada en este proyecto se podría utilizar para la detección de otras características, tales como barbilla, cejas, orejas, etc.

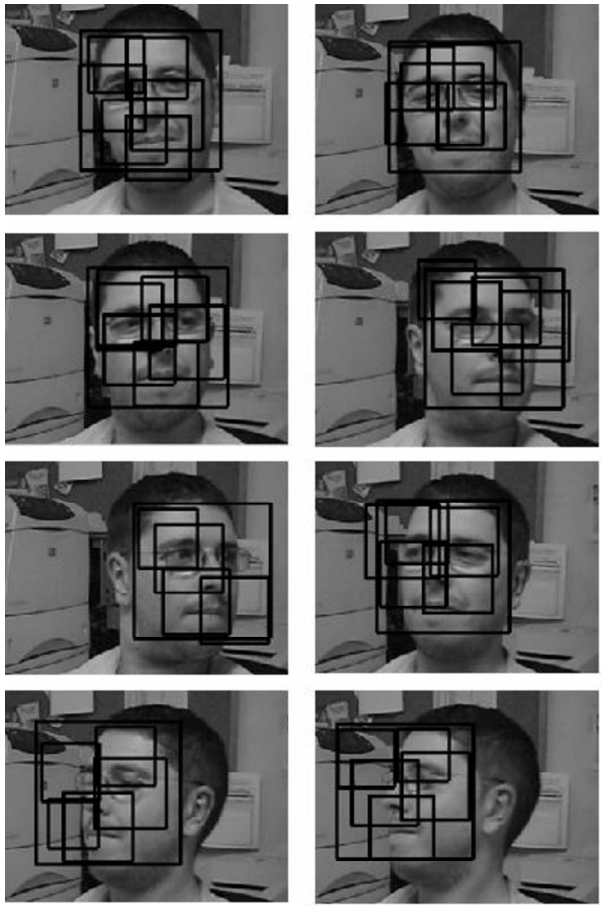


Figura 33. Detección de características faciales.

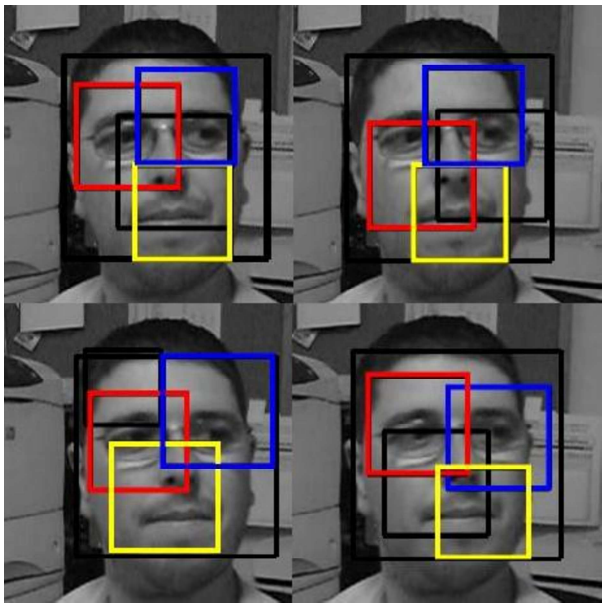


Figura 34. Detección y seguimiento de características faciales.

7. Planificación

En la tabla siguiente veremos la planificación temporal del desarrollo de nuestro proyecto, remarcando que se ha extendido a lo largo de una año universitario. Podemos ver como se han ido consiguiendo poco a poco todas las espectativas iniciales.

Lectura Bibliográfica	40h
Estudio de viabilidad	20h
Detección de caras	
Creación de proyecto en blanco	2h
Implementación FaceDetect	5h
Modificaciones necesarias sobre el código	5h
Estudio de software necesario para edición de vídeo	3h
Depuración de código	15h
Pruebas	20h
Entrenamiento Cascada	
Implementación código para recortar características faciales	5h
Generación de código para entrenar cascadas específicas	10h
Entrenamiento de cascadas específicas de características faciales	2h
Pruebas	15h
Implementación CSR	
Búsqueda información de CSR escala de grises	3h
Búsqueda información de CSR orentaciones	7h
Estudio de viabilidad y mejoras aplicando CSR orientaciones	15h
Implementación CSR	30h
Depuración de código	10h
Pruebas	25h
Implementación e integración del SIFT en la fase de Tracking	
Búsqueda información del descriptor SIFT	5h
Estudio de viabilidad y de posibles adaptaciones	5h
Implementación de Pseudo-SIFT	15h
Depuración de código	10h
Pruebas	10h
Redacción de la memoria	40h
TOTAL	**317h**

8. Conclusiones

En este trabajo se han analizado fotos y secuencias de vídeo en entornos no controlados. El desarrollo se ha centrado en la detección y seguimiento de características que faciliten a la visión robótica la interacción hombre-máquina. El objetivo inicial era detectar la existencia o no existencia de un individuo en una secuencia de frames, y a partir de las detecciones positivas analizar su contenido, como las caras y características internas faciales, o seguir la evolución de las mismas con tal de extraer información de forma automática. A partir de formatos de entradas multimedia (fotos), el primer paso consiste en detectar caras mediante aprendizaje estadístico y detectarlas mediante Adaboost usando una cascada de clasificadores. Estos clasificadores trabajan sobre las características Haar-like, que nos han permitido realizar aprendizajes y detecciones robustas aún con las transformaciones típicas que nos podemos encontrar, tales como cambios de iluminación o transformaciones en la forma de los objetos a detectar. Una vez la cara es detectada se lanza el método de detección de características faciales. Por último, se ha introducido un nuevo algoritmo llamado Pseudo-SIFT, que se basa en el descriptor original SIFT para extraer las características más relevantes de un objeto y ser detectado de forma robusta. El sistema final se ha optimizado para procesar los frames en un periodo cercano a tiempo real. Los análisis también muestran que los datos del sistema pueden ser usados para complementar otras técnicas del área, tales como modelar la detección, comportamiento, o interacción de personas de forma automática. De igual forma, su utilidad podría ampliarse a la interacción hombre-máquina sobre robots programables o para sistemas de auto-vigilancia.

Anexo 1: Visión Artificial Aplicada al Sistema de Visión Humano.

Como ya se ha comentado al principio de este documento, la Visión por Computador es una sub-área de la Inteligencia Artificial y, como vemos en el esquema de la figura .1, esta relacionada con diversas áreas afines.

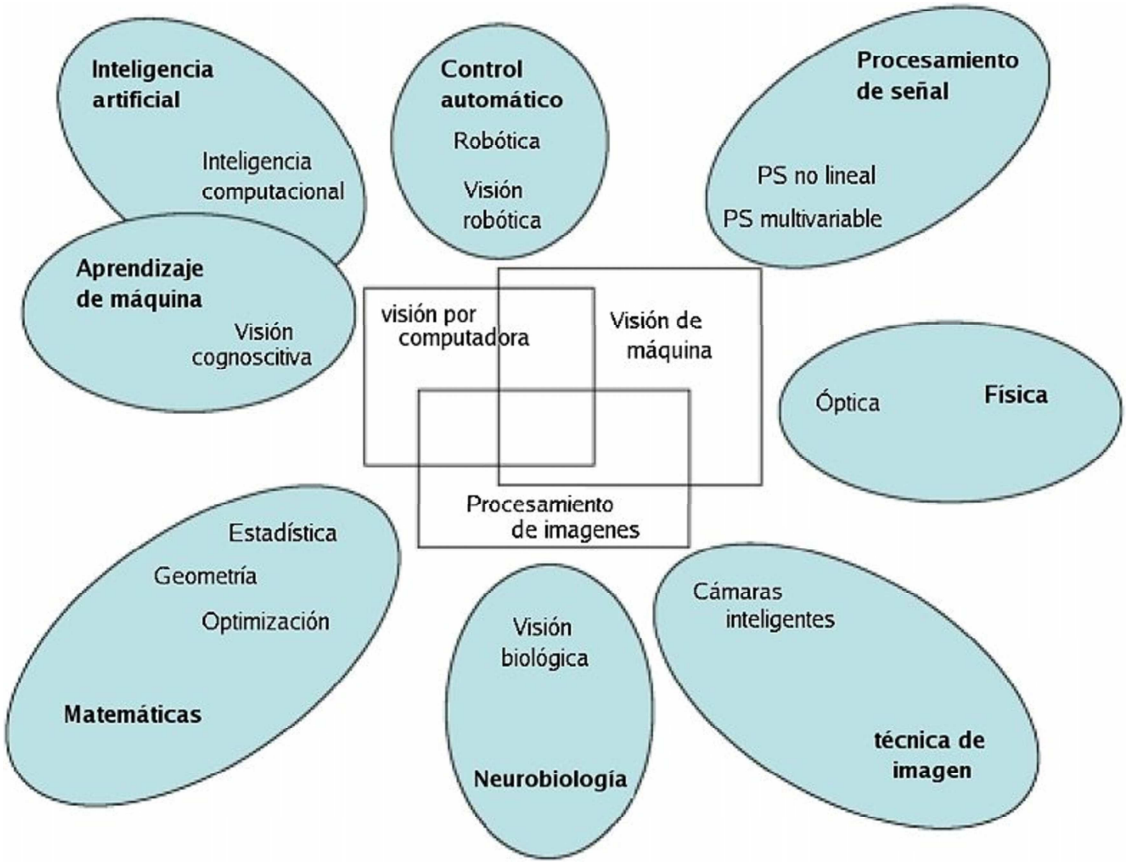


Figura .1. Esquema de relaciones de la visión por computador y otras áreas afines

Algunos de los objetivos de la Visión Artificial son los siguientes:

- La detección, segmentación, localización y reconocimiento de ciertos objetos en imágenes (ej, caras humanas).
- La evaluación de los resultados (ej.: segmentación, registro).
- Registro de diferentes imágenes de una misma escena u objeto, ej, hacer concordar un mismo objeto en diversas imágenes.
- Seguimiento de un objeto en una secuencia de imágenes.

- Mapeo de una escena para generar un modelo tridimensional de la escena; tal modelo podría ser usado por un robot para navegar por la escena.
- Estimación de las posturas tridimensionales de humanos.
- Búsqueda de imágenes digitales por contenido.

Estos objetivos se consiguen por medio de reconocimiento de patrones, aprendizaje estadístico, geometría de proyección, procesado de imágenes, teoría de gráficos y otros campos. La Visión Artificial cognitiva está muy relacionada con la psicología cognitiva y la computación biológica.

Actualmente, uno de los proyectos que se están potenciando es la Visión Artificial aplicada al Sistema Visual Humano.

¿Qué es el sistema visual? La visión es un sentido que consiste en la habilidad de detectar la luz y de interpretarla. La visión es propia de los animales, teniendo éstos un sistema dedicado a ella llamado sistema visual. La visión artificial extiende la visión a las máquinas.

La primera parte del sistema visual se encarga de formar la imagen óptica del estímulo visual en la retina (sistema óptico). Esta es la función que cumplen la córnea y el cristalino del ojo.

Las células de la retina forman el sistema sensorial del ojo. Las primeras células en intervenir son los fotorreceptores, los cuales capturan la luz que incide sobre ellos. Se dividen en dos tipos: los conos y los bastones. Otras células de la retina se encargan de transformar dicha luz en impulsos electroquímicos y en transportarlos hasta el nervio óptico. Desde allí, se proyectan a importantes regiones cerebrales como el núcleo geniculado lateral y la corteza visual.

Figura .2. Ojo Humano.

En el cerebro comienza el proceso de reconstruir las distancias, colores, movimientos y formas de los objetos que nos rodean.

Cuando la retina está dañada o no funciona bien, los fotorreceptores dejan de funcionar, pero eso no quiere decir que toda la estructura del Sistema Visual Humano no pueda seguir funcionando. Por ello hay científicos que están desarrollando microchips de silicio que puedan dotar de visión artificial a

aquellas personas a las que no les funcionan los fotorreceptores. La información captada por los fotorreceptores se transmite a las células ganglionares, donde se interpreta y se manda al cerebro a través del nervio óptico. Existen enfermedades que afectan a estas células, como la retinitis pigmentaria o la DMAE, que dejan inoperativos los fotorreceptores pero no dañan las células ganglionares o el nervio óptico, con lo cual el problema no es que la información no pueda llegar al cerebro, sino que no se puede captar. En estos casos se pueden desarrollar unos conos y bastones artificiales. Los requisitos de los microchips para que cumplan la función de los fotorreceptores son: que sean lo suficientemente pequeños como para implantarlos en el ojo, y que tengan una fuente de abastecimiento de energía continua. Que no causen rechazo, es decir, que sean biocompatibles con los tejidos del ojo. Uno de los micros que se ha desarrollado con éxito por el momento es un dispositivo de 2mm de diámetro y fino como un pelo humano. Contiene 3500 células solares microscópicas que imitan a los bastones y los conos y convierten la luz en pulsos eléctricos. Se abastece de energía solar, con lo que se evitan cables y baterías.

La Visión Artificial es la adquisición automática de imágenes sin contacto y su análisis también automático con el fin de extraer la información necesaria para controlar un proceso o una actividad como control de calidad, ordenación por calidades (grading), manipulación de materiales test y calibración de aparatos, monitorización de procesos, etc.

Paralelamente, la ciencia también está investigando una manera de devolver la visión a gente ciega. De momento las pruebas se están realizando sobre roedores, pero se cree que podrá llegar a aplicarse en los humanos. El método consiste en transplantar fotorreceptores.

Un equipo de investigadores ha demostrado que la retina se puede regenerar y recuperar la visión perdida. Lo han logrado con el trasplante de un tipo de fotorreceptores inmaduros, células de la retina sin las cuales resulta imposible transmitir las señales visuales al cerebro. El implante de esas células logró regenerar los fotorreceptores dañados y restaurar la visión en un experimento con ratones que se publicó en la revista *Nature*. De momento, es sólo un tratamiento experimental con roedores, pero la nueva estrategia se perfila como una cura potencial para cegueras intratables y tan comunes como las que ocasionan la diabetes o la degeneración macular asociada a la edad.

Se están realizando estudios de este tipo desde hace años, buscando una fórmula para regenerar las retinas dañadas, con los fotorreceptores como diana del tratamiento. La retina es una porción del sistema nervioso central que está formada por varias capas celulares organizadas para transmitir las señales visuales al cerebro. La principal causa de daño en esta zona es el deterioro de los fotorreceptores. Por ello, los esfuerzos se han encaminado a regenerar estas células, al igual que la Inteligencia Artificial está buscando sus propios

métodos, basándose en la misma ideología.

Restaurar las células fotorreceptoras debería ser relativamente sencillo porque la mayoría de las conexiones en el cerebro que permiten la visión permanecen intactas en estos pacientes. Pero, pese a los intentos con implantes de células madre, aún no han tenido éxito. Se pensaba que las células madre implantadas no lograban desarrollarse por la propia retina, que actuaba como un entorno hostil que inhibía la regeneración de los fotorreceptores. Los científicos de la Universidad de Michigan (EE.UU.) y del Colegio Universitario de Londres han demostrado que ése no era el problema.

La nueva estrategia también recurre al trasplante celular, aunque en un momento de su desarrollo más avanzado. En este caso se utilizaron células inmaduras de bastones (fotorreceptor que permite la visión nocturna), cuando ya habían dejado de dividirse. La clave de la investigación está en el hallazgo de una ventana de oportunidad durante la cual esas células se han convertido ya en bastones, pero todavía no han empezado a funcionar como tales.

A. Anexo 2: La percepción y la visión pre-atentiva.

B. La percepción.

En cualquier método de percepción existen dos fases, la *pre-atentiva* y la *construcción personal o atentiva.*

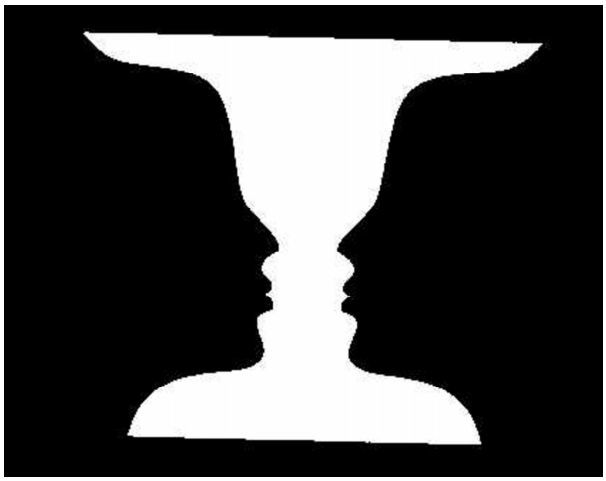


Figura B.1. Percepción.

La percepción de puede definir como la sensación interior resultante de una impresión material, hecha por los sentidos. Para la psicología, la percepción es uno de los procesos cognoscitivos, una forma de conocer el mundo.

Actualmente consideramos que la percepción es un proceso cíclico, de carácter activo, constructivo, relacionado con procesos cognitivos superiores y que transcurre en el tiempo. La percepción es un proceso complejo que depende tanto de la información que el mundo entrega, como de la fisiología y las experiencias de quien percibe; éstas afectan tanto el acto perceptivo mismo, por la alteración de los esquemas perceptivos, tanto como a otros procesos superiores, como son las motivaciones y las expectativas. Por ejemplo, dependiendo de quién y cómo vea la imagen B.1, podría ver una columna o dos caras.

Como ya dijimos, el acto perceptual se considera cíclico. Este ciclo constaría de dos fases: en la primera, denominada preatentiva, el individuo detecta la información sensorial y la analiza; en la segunda fase, denominada construcción personal, se produce el objeto perceptual específico. En el acto perceptivo se da una constante anticipación de lo que sucederá, basada en información que acaba de ingresar a los órganos de los sentidos y en esquemas, patrones que seleccionan la información a procesar en base a criterios probabilísticos extraídos de la experiencia previa, los cuales son modificados a su vez por la nueva experiencia perceptiva, y que dirigen los movimientos y las actividades

exploratorias necesarias para obtener más información. Como los esquemas son modificados tras cada experiencia perceptiva y éstos determinan que información sensorial se procesará y cuales serán los patrones de búsqueda para obtenerla, las siguientes experiencias perceptivas tendrán la influencia de las anteriores percepciones, no existiendo la posibilidad que dos experiencias perceptuales sean idénticas.

B.1. El proceso de selección de información.

B.1.1. Análisis de la información sensorial.

La información que entra a los sentidos es abstraída por los mecanismos analizadores. Por ejemplo, a nivel visual los analizadores extraen informaciones sobre color, realzan contornos, determinan la dimensión y la dirección del movimiento de las imágenes visuales; a nivel de audio extraen la altura y el volumen de las imágenes acústicas y determinan las relaciones espaciales y temporales entre las señales visuales y las acústicas.

B.1.2. El reconocimiento de pauta pasivo y activo.

El reconocimiento de pauta podría establecerse de modo pasivo como activo.

El reconocimiento pasivo constituiría una ejecución automática y eficiente del análisis de los hechos sensoriales entrantes. De este modo, se reducen las posibilidades perceptuales a un número pequeño, de fácil manejo.

El reconocimiento activo consistiría en hacer coincidir propiedades sintetizadas internamente con las señales analizadas; esta permitirá detectar la información ambigua, incorrecta o faltante. Este proceso permitiría, de acuerdo al significado, al contexto y las expectativas del material ya presentado, reducir a un número pequeño de señales el conjunto de estímulos a analizar.

B.2. La percepción visual

Según Bayo, citando la teoría de Gibson, la percepción visual dependería de la *estimulación ordinal*, o sea, a la disposición particular de los rayos luminosos que inciden en la retina humana, ya que el organismo no puede responder a la dirección y al carácter de los rayos como tales, ya que cuando la energía luminosa que incide sobre la retina es uniforme la percepción no existe [13]. Así, la percepción de borde o contorno se daría por la existencia de un salto de elementos, del tipo 'cccccooooo' (c=claro;o=obscuro); la textura se da con un

formato 'cccooocccooo' y un gradiente de textura como 'coooccoooccccooocccccooo'. De este modo, la fotografía con división de colores, si bien a la cercanía aparece como un conjunto aleatorio de puntos de colores, a la distancia se percibe con las mismos contornos y texturas del original, ya que nuestros ojos no responden a la estimulación luminosa en sí, sino al orden en que esta se encuentra. Los analizadores serían las entidades que reconocen estas pautas y las envían a la memoria sensorial, la cual enviará la información a la consciencia de acuerdo a la pertinencia de la información.

C. La visión preatentiva

Centrándonos un poco más en la visión, un aspecto importante sobre la forma o mecanismo que tenemos de *construir* la imagen o escena visual es el hecho de que cuando abrimos los ojos, antes de que se inicien los mecanismos atentivos y de foveolización, de una forma inconsciente ya percibimos ese mundo real según figuras, no tenemos un primer estadio de composición de la escena externa según bordes, contrastes, puntos con distinto brillo o color. Directamente tenemos un primer nivel donde ya aparecen las figuras, formas y objetos en general. Cabría preguntarse si esto es compatible con una teoría constructuvista de la visión.

Existen diferentes aproximaciones para intentar dar respuesta a la cuestión anterior, en general se producirán dos estadios diferentes hasta la situación final de la visión. Primero habría un nivel o estadio preatentivo de procesamiento de ciertos elementos *primitivos*, y luego se produciría un segundo nivel o estadio atentito, consciente y con carácter cognitivo, donde el sujeto agrupa esos elementos primitivos en ciertas formas unitarias que pueden ser superficies, manchas de color, texturas, etc. o incluso objetos. Este segundo estadio se debería a la actividad gestáltica.

Ann Treisman [10] defiende esta idea, considerando que el elemento primitivo base serían las texturas, el textón, como unidad elemental. Irving Biederman [11] propone que el elemento base es la estructura geométrica tridimensional, o geón, tal como refiere D. Marr. La combinación de geones conforma el objeto. Esta idea, con cierto carácter constructivista, entra en lo que denominamos línea *bottom up*, donde la imagen final se alcanza desde un proceso de nivel inferior, construida con elementos tipo figura, pero en todo caso, más simples.

Frente a esta concepción de orden constructivista se sitúan los defensores puros de la Gestalt. Para ellos en la fase preatentiva, inconsciente, ya aparecen figuras, como se señaló anteriormente, al abrir los ojos, incluso en el momento de no atención, vemos cosas definidas y, en la fase atentiva, buscamos objetos o puntos de fijación donde focalizamos nuestra atención. Es un mecanismo

de tipo *top down*, a diferencia de la anterior, constructivista, que elabora la información de abajo hacia arriba, *bottom up*.

Parece lógico pensar que ambas aproximaciones tienen parte de razón. El análisis de la estructura neurológica del sistema visual, desde la retina hasta el córtex, hace pensar en un proceso constructivista. Se tomarían texturas u otro tipo de información, pero habría una cierta descomposición del mundo exterior para luego recomponerlo en base a patrones predefinidos, donde estos patrones o moldes tendrían un origen evolutivo fruto del aprendizaje o del habito.

Anexo 3: CD.

A. Contenido del CD.

- /Ejecutables: Contiene los archivos ejecutables del sistema.
- /Código: Contiene las fuentes del sistema.
- /Datos: Contiene imágenes y vídeos de las bases de datos utilizadas y generadas, así como las bases de datos públicas usadas en los experimentos.
- /Memoria: Contiene el archivo PDF con esta memoria.

Referencias

[1] Paul Viola & Michael Jones, 'Robust Real-time Object Detection', International Journal of Computer Vision, 2002.

[2] J. Friedman & T. Hastie and R. Tibshirani, 'Additive logistic regression: a statistical view of boosting', Stanford University, Technical Report, 1998.

[3] Timor Kadir, 'Saliency, Scale and Image Description', International Journal of Computer Vision, pp. 83-105, 2004.

[4] Lowe, D. G., 'Distinctive Image Features from Scale-Invariant Keypoints', International Journal of Computer Vision, 60, 2, pp. 91-110, 2004

[5] G. Yang and T. S. Huang, 'Human Face Detection in Complex Background', Pattern Recognition, vol. 27, no. 1, pp. 53-63, 1994

[6] M.-H.Yang, D. Kriegman and N. Ahuja, 'Detecting Faces in images: a Survey', IEEE Transactions on Pattern Analysis and Machine Intelligence 24, n°1, pp.34-58, 2002

[7] H. A. Rowley, S. Baluja, and T. Kanade, 'Neural network-based face detection', IEEE Transactions on Pattern Analysis and Machine Intelligence. 20, January 1998, 23-38

[8] M. A. Turk & A. P. Pentland, 'Face recognition using eigenfaces', Proceedings of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp. 586-591, Maui, Hawaii 1991

[9] Viola P. & Jones M.: Rapid Object Detection using a Boosted Cascade of Simple Features, Computer Vision and Pattern Recognition, 2001

[10] Treisman AM. 'Features and Objects in visual processing'. Scientific American, 1986, Nov. 106-115.

[11] Biederman I. 'Recognition by components: A theory of human image understanding'. Psychol. Rev. 1987, 94: 115-147.

[12] K. Mikolajezyk and C. Schmid. Affine invariant interest point detectors. International Journal of Computer Vision, 60:63-86, 2004.

[13] Bayo, J. (1987). Percepción, desarrollo cognitivo y artes visuales.

[14] Marr, 1982 Marr, D. (1982). Vision. H. Freeman and Co.

[15] T. Kadir, and M. Brady, 'Saliency, Scale and Image Description', International Journal of Computer Vision, vol. 45, issue 2, pp-83-105, 2001.

[16] P.J. Flynn, 'Saliencies and symmetries: Toward 3D object recognition from large model databases', Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 322-327, 1992.

[17] B.Schiele and J. L. Crowley, 'Probabilistic object recognition using multidimensional receptive field histograms', Proceedings of the International Conference in Pattern Recognition, Vienna, Austria, 1996.

[18] N. Sebe and M.S. Lew, 'Salient points for content-based retrieval', Proceedings of the British Machine Vision Conference, pp. 401-410, 2001.

[19] U. Neisser, 'Visual Search', Scientific American, vol. 210, issue 6, pp. 94-102, June 1964.

[20] J. Matas, O. Chum, M. Urban, and T. Pajdla, 'Robust Wide baseline Stereo from Maximally Stable Extremal Regions', Proceedings of the British Machine Vision Conference, vol. 1, pp. 384-393, 2002.

[21] C. Schmid and R. Mohr, 'Local gray value invariants for image retrieval', IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 19, issue 5, pp. 530-535, 1997.

[22] J. Sivic and A. Zisserman, 'Video google: A text retrieval approach to object matching in videos', Proceedings of the International Conference on Computer Vision, Nice, France, 2003.

[23] R. Fergus, P. Perona, and A. Zisserman, 'Object class recognition by unsupervised scale-invariant learning', Proceedings IEEE Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA, 2003.

[24] N. Otsu, ."A Threshold Selection Method for Gray Level Histograms", *IEEE transactions on SMC*, 1979.

[25] F. Crow, 'Summed-area tables for texture mapping', In Proceedings of SIGGRAPH, volume 18(3), pp. 207-212, 1984.

[26] Robert E. Schapire, Yoav Freund, Peter Bartlett, and Wee Sun Lee, 'Boosting the margin: A new explanation for the effectiveness of voting methods', In Proceedings of the Fourteenth International Conference on Machine Learning, 1997.

[27] R.Lienhart and J.Maydt, 'An extended set of haar-like features for rapid object detection', Proc. of the IEEE Conf. On Image Processing, pp. 155-162, 2002.

[28] Haffner, and Yann Le Cun. 'Boxlets: a fast convolution algorithm for signal processing and neural networks', In M. Kearns, S. Solla, and D. Cohn, editors, Advances in Neural Information Processing Systems, volume 11, pp. 571-577, 1999.

[29] Koenderink, J.J., 'The structure of images. Biological Cybernetics', pp:363-396, 1984.

[30] Lindeberg, T., 'Scale-space theory: A basic tool for analysing structures at different scales', Journal of Applied Statistics, 21(2):224-270, 1994.

[31] David Lowe, 'Distinctive Image Features from Scale-Invariant Keypoints', 2004.

[32] E. Shanon, A Mathematical Theory of Communication, 1948.

[33] URL: floriolguin.blogspot.com/2007/07/la-vision-artificial.html

[34] URL: www.retinamadrid.org/noticias/1

[35] URL: www.robertexto.com/archivo1/percepcion.htm

[36] URL: www.caltech.edu/