

Aplicación de los diferentes espacios de color para detección y seguimiento de caras.

C. Pérez , M. A. Vicente, C. Fernández, O. Reinoso, A. Gil
Universidad Miguel Hernández
División Ingeniería de Sistemas y Automática
Departamento de Ingeniería de Sistemas Industriales, Campus de Elche
Avda. del Ferrocarril s/n 03202 Elche (Alicante) España
suni@umh.es

Resumen

El objetivo de este artículo es presentar el esquema de un sistema completo de visión que realiza las tareas de detección, reconocimiento y seguimiento de una cara humana. Se analizan los espacios de color empleados para realizar las tareas antes mencionadas, se comparan dos técnicas de reconocimiento de caras basadas en la proyección de las imágenes en diferentes subespacios (PCA e ICA) y se desarrollan dos técnicas de tracking, una basada en modelo y otra basada en características.

Palabras Clave: detección de caras, tracking, reconocimiento de caras, ICA, PCA

1 INTRODUCCIÓN

En el presente artículo se describirá con detalle un sistema de visión completo que realiza las tareas de detección y reconocimiento de caras humanas y el seguimiento de la cara reconocida. Para realizar las tareas anteriores se utilizan técnicas de detección del color de la piel, éste puede variar según el tipo de etnia al que pertenezcan los individuos y/o la iluminación del entorno, por lo que se utilizan técnicas de detección y reconocimiento basadas en la apariencia para generalizar el modelo del color de piel humana.

En la Figura 1 se muestra un esquema del algoritmo de visión completo.

En los siguientes apartados se desarrollaran cada una de las etapas propuestas en el esquema anterior.

2 DETECTORES DE CARAS

2.1 INTRODUCCIÓN

La detección automática de caras humanas en imágenes ha sido objeto de investigación durante décadas. Recientemente dos artículos complementarios revisan el estado del arte actual de la detección de caras [20][9]. Los diversos métodos de detección de caras pueden clasificarse en dos grandes grupos: los métodos tradicionales basados

en detección de características (“*featured-based*”), que utilizan datos explícitos como modelos colorimétricos o geométricos de caras y los métodos más recientes que se basan en métodos de reconocimiento de patrones (“*image-based*”), que obtienen la información de manera implícita mediante el aprendizaje desde ejemplos o patrones. Los métodos basados en detección de características hacen uso de características de bajo nivel como son los bordes, los niveles de gris, el color y el movimiento. La distribución de mínimos locales de niveles de gris puede señalar la presencia de cejas, pupilas y labios y por lo tanto marcar la presencia de una cara en la imagen. En un espacio de crominancia normalizado, el color de la piel (sea cual sea el grupo étnico al que pertenezca la cara a detectar) puede ser modelizado mediante una distribución gaussiana sencilla. Características faciales de mayor resolución pueden ser buscadas secuencialmente, comenzando con los ojos, o por grupos mayores de características. La mayoría de las técnicas anteriores se utilizan sólo para detección de cabezas y hombros y en escenas frontales.

Para resolver problemas más complejos como la detección de múltiples caras en fondos no controlados se utilizan los detectores basados en reconocimiento de patrones o aprendizaje de ejemplos, sin utilizar una formulación explícita para la cara. La mayoría de estos métodos requieren como paso preliminar un costoso proceso de búsqueda por ventanas a diferentes resoluciones (diferentes escalas y posiciones). Estos detectores se basan en redes neuronales artificiales [16] que distinguen entre entradas que sí son caras de las que no lo son.

Los métodos basados en características son apropiados para procesamiento de imágenes en tiempo real cuando el color y el movimiento son posibles. Para imágenes en niveles de gris estáticas, los métodos basados en reconocimiento de patrones son más adecuados.

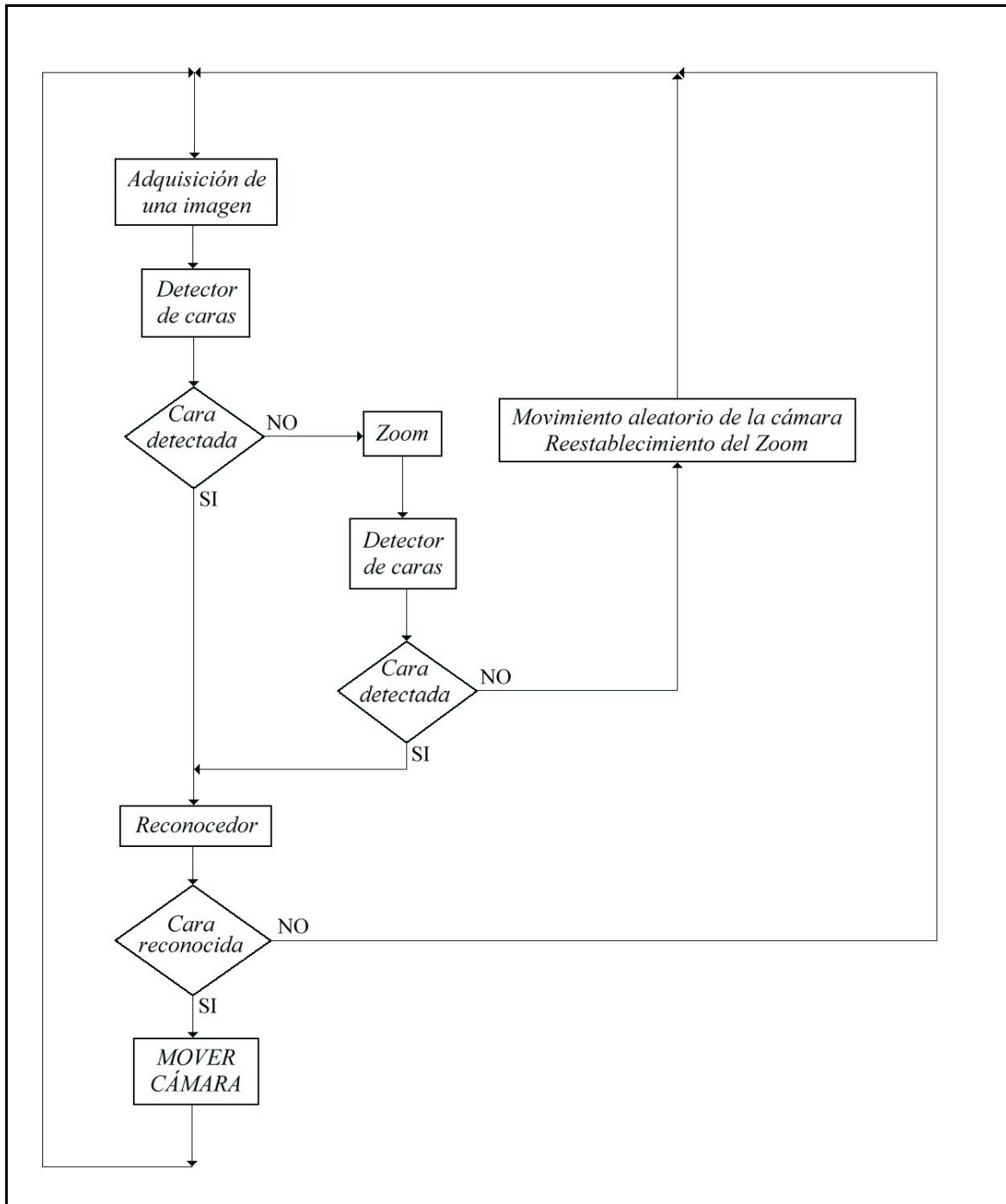


Figura 1. Esquema del algoritmo de visión.

2.2. DETECTOR DE CARAS BASADO EN COLOR Y REDES NEURONALES.

En nuestro sistema de visión se ha empleado para la detección de caras un método de reconocimiento de patrones que detecta la presencia de caras mediante el aprendizaje del color de la piel, por lo que podría considerarse un detector mixto según la clasificación del apartado anterior. El algoritmo utilizado se basa en un clasificador basado en redes

neuronales que distingue los colores naturales asociados a las caras de los que no lo son.

La red neuronal empleada es el perceptrón multicapa con entrenamiento hessiano, basado en el algoritmo de Levenberg-Marquardt [7].

Presentando a la capa de entrada de la red neuronal diversas imágenes de caras y no caras (ver Figura 2) se ha entrenado el sistema hasta lograr la convergencia de los pesos y minimizar el error.



Figura 2. Algunos ejemplos de imágenes de caras y no-caras utilizadas durante el entrenamiento.

Para comprimir la información contenida en las imágenes de las caras y no caras, se ha utilizado la combinación de los histogramas de la imagen en cada uno de los canales de color como vector de entrada a la red. Se han probado diferentes espacios de color (ver Apéndice) durante el entrenamiento, consiguiendo las mejores tasas de detección para el espacio de color YES, tal como se muestra en la Tabla 1.

| espacio de color | % aciertos |
|------------------|------------|
| RGB | 75% |
| YES | 95% |
| YIQ | 80% |
| YUV | 81% |
| YCbCr | 79% |

Tabla 1. Resultados del entrenamiento de la red con diferentes espacios de color

Una vez que el clasificador neuronal está entrenado se utiliza como detector de posibles caras dentro una escena global, para ello, primero hay que realizar un paso previo que consiste en muestrear la imagen original con ventanas de diferente tamaño y posición, de cada ventana se calculará su histograma que pasará como entrada al clasificador y éste señalará las posibles caras dentro de la imagen. Como puede ocurrir que se obtengan ventanas solapadas hay que unir estas ventanas y quedarse con la mayor de todas, además hay que cuidar que tengan unas medidas correctas para evitar falsos positivos, esto se realiza mediante el filtrado de los rectángulos-ventanas que se consideren demasiado grandes o demasiado pequeños o sean muy estrechos horizontal o verticalmente. En la Figura 3 se muestran los resultados del algoritmo propuesto sobre una

imagen, se consiguen detectar las caras pero también otras zonas de color similar.

Para eliminar estos falsos positivos (las manos en la Figura 3.) es posible utilizar otro clasificador supervisado para eliminarlos, en nuestro caso no los hemos filtrado y se consideran como posibles caras a reconocer que han de descartarse durante la etapa de reconocimiento de la cara. Por lo tanto, las ventanas detectadas como caras en esta fase de detección pasan a la siguiente fase de reconocimiento del individuo presente en la imagen.

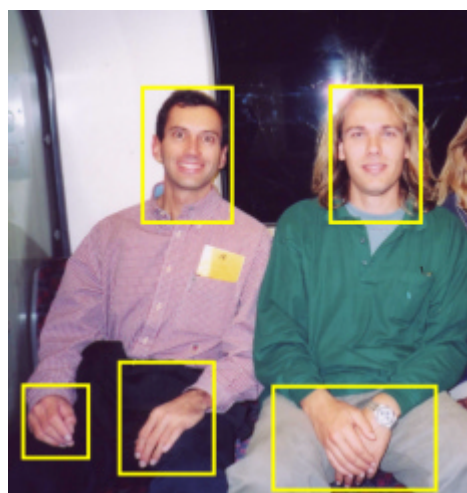


Figura 3. Resultado de la aplicación del detector de caras sobre una escena.

2. RECONOCIMIENTO DE CARAS

En los últimos años se han realizado progresos considerables en la detección y reconocimiento de caras, especialmente en el proceso denominado “mug shots”, que son fotografías dónde solo se ve la vista frontal de la cabeza del sujeto a reconocer tomadas con iluminación controlada y siempre con la misma escala.

Los mejores resultados han sido obtenidos en 2D utilizando técnicas basadas en el análisis de características ([1],[3]), técnicas basadas en las eigencaras (o eigenimágenes) ([12],[17],[19]), o bien combinando ambas técnicas ([6],[13],[14]).

En el análisis de características se pretende reconocer al sujeto a partir de ciertas características extraídas de la imagen, como la distancia entre los ojos, la longitud de la boca, etc.

En las técnicas basadas en eigenimágenes se cambia el espacio original de la imagen por otro espacio generado a partir de las imágenes originales, esto se realiza haciendo uso de la transformada de Karhunen-Loeve y también se

conoce como PCA¹. Existen otros espacios diferentes donde realizar posteriormente el reconocimiento, y son el generado mediante un análisis de discriminantes lineales (LDA) [22], o bien el espacio generado tras aplicar técnicas de ICA² [11],[5], estas técnicas aunque más novedosas proporcionan peores tasas de reconocimiento que las técnicas clásicas de PCA como veremos en el apartado de resultados.

2.1. Sistema de reconocimiento basado en proyecciones

Dentro de nuestro sistema de visión hemos utilizado para el reconocimiento e identificación de las caras el método basado en la proyección de las imágenes originales en un nuevo espacio.

El algoritmo básico para identificar caras (mug shots) en diferentes subespacios es el siguiente:

1. Se calcula un subespacio (generado por las propias imágenes de las caras a reconocer) donde proyectar las imágenes.
2. Un vez generado el subespacio, todas las caras de entrenamiento son proyectadas en el subespacio
3. Cada nueva imagen se proyecta en el subespacio y es comparada con todas las proyecciones de la imágenes de entrenamiento por similitud o por medida de distancias, la imagen de entrenamiento más similar o más cercana a la nueva imagen se utiliza para identificar esa nueva cara.

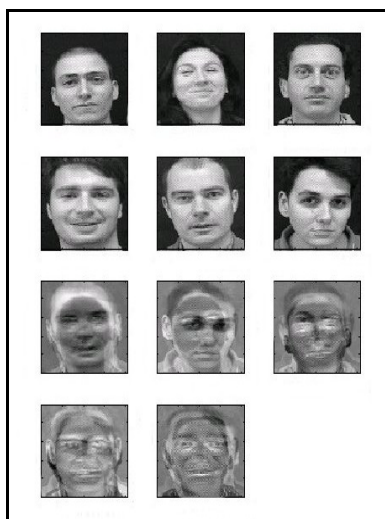


Figura 4. Imágenes de entrenamiento y eigenvectores obtenidos con PCA1.

2.1. Reconocimiento mediante PCA e ICA

Se han comparado dos métodos basados en proyecciones sobre subespacios para su utilización en el reconocimiento de caras: PCA e ICA.

Dos algoritmos distintos se han utilizado para obtener el subespacio PCA [18], en uno de ellos, que lo denotamos como PCA1, sólo se calcula un subespacio por cada cara representativa y en el otro, PCA2, se obtienen tantos subespacios como caras a identificar. Para ambos subespacios se han hecho pruebas con diferentes medidas de distancias (Mahalanobis y las normas L2 y L1) y para el caso del subespacio PCA1 se ha utilizado además un perceptrón multicapa como clasificador, obteniendo los mejores resultados de reconocimiento. También se han realizado experimentos con diferentes tipos de imágenes generadoras del subespacio (originales y bordes) tal como se resume en la Tabla 2. Como se puede observar en la tabla, los mejores porcentajes de aciertos se obtienen para PCA1 utilizando un MLP como clasificador y con el subespacio PCA2 generado a partir de imágenes de bordes y utilizando la distancia de Mahalanobis. La diferencia computacional entre ambos algoritmos es considerable, siendo mucho más rápido la utilización del subespacio PCA1 junto con el MLP que cualquier combinación de medida de distancia y PCA2 .

| algoritmo | imagenes | distancia | % aciertos |
|-----------|------------|------------------|------------|
| PCA1 | originales | L2 | 59% |
| PCA1 | originales | Mahalanobis | 66% |
| PCA1 | originales | MLP ³ | 95% |
| PCA1 | bordes | L2 | 30% |
| PCA1 | bordes | Mahalanobis | 54% |
| PCA1 | bordes | MLP | 60% |
| PCA2 | originales | L2 | 85% |
| PCA2 | originales | Mahalanobis | 34% |
| PCA2 | bordes | L2 | 45% |
| PCA2 | bordes | Mahalanobis | 95% |

Tabla 2. Tasas de reconocimiento con PCA

La generación del subespacio mediante ICA se ha realizado utilizando el algoritmo FastICA (version 2.1 para Matlab 5.X) descrito con detalle por Hyvärinen en [11], y los resultados obtenidos se muestran en la Tabla 3. Se puede observar como las tasas de reconocimiento son algo inferiores a las obtenidas con PCA1 y PCA2. Por lo tanto, el mejor

¹ PCA, Análisis de Componentes Principales

² ICA, Análisis de Componentes Independientes

³ MLP, perceptron multicapa como clasificador

algoritmo en cuanto a tasa de reconocimiento y rapidez de ejecución que se implementó para la etapa de reconocimiento es el subespacio PCA1 junto el clasificador neuronal.

| algoritmo | imagenes | distancia | % aciertos |
|-----------|------------|-------------|---------------|
| FastICA | originales | L2 | 54% |
| FastICA | originales | Mahalanobis | 63% |
| FastICA | bordes | L2 | 22% |
| FastICA | bordes | Mahalanobis | 49% |

Tabla 3. Tasas de reconocimiento con ICA (FastICA)

3. ALGORITMOS DE TRACKING

3.1. INTRODUCCIÓN

Atendiendo al tipo de procesamiento de la imagen, podemos clasificar los algoritmos de seguimiento o *tracking* como basados en modelo y basados en características.

Los sistemas de *tracking* basados en modelo se fundamentan en la búsqueda en la imagen de un modelo más o menos concreto del objeto a seguir. El modelo del objeto, puede ser una representación CAD o como en nuestro caso el modelo más abstracto de una cara reconocida por alguna de las técnicas presentadas. El resultado en cualquier caso es la posición del objeto/cara en la imagen adquirida.

Los sistemas de *tracking* basados en características a pesar de ser menos robustos, suelen ser menos costosos computacionalmente, por lo que el tiempo de ejecución desciende en gran medida incrementándose la velocidad de seguimiento.

En este artículo se han comparado dos algoritmos que emplean estas técnicas (seguimiento basado en modelo y en características).

Para el primer caso (basado en modelo), hemos implementado el flujograma de la Figura 1, en el que en cada iteración se hace el procesado de la imagen para detectar y reconocer las caras. Esta información se utiliza como entrada del regulador PID que controla los movimientos de la cámara motorizada (en nuestro caso una SONY EVID-31), (ver Figura 5).

En el segundo caso hemos implementado un algoritmo basado en características cuyo esquema de control corresponde al de la Figura 6, en el que

podemos ver que la etapa llamada “DETECCIÓN + RECONOCIMIENTO” se ha sustituido por “*Camshift*” (que describiremos a continuación), y que a grandes rasgos se basa en el seguimiento del color de la piel humana.

Al tratarse de un seguimiento basado en color de la piel, no será capaz de discriminar entre piel de la cara o de otras partes del cuerpo (manos, hombros, ...) ni entre diferentes caras, por lo que será mucho menos robusto pero ante la presencia de pocos elementos de color piel, seguirá con mucha mayor eficacia al objeto de mayor tamaño.

3.2. TRACKING BASADO EN EL COLOR DE LA CARA (*tracking* basado en características)

3.2.1. Introducción

Los sistemas de *tracking* de caras tanto los basados en modelo, como en características, se pueden agrupar en tres grandes tipos: (1) *tracking* 2D, realizan seguimiento solamente de la posición de la proyección de la cara en el plano de imagen [2]; (2) *tracking* 2 D, se realiza seguimiento 2D con alguna información de orientación [8]; (3) *tracking* 3D, se realiza el seguimiento de la posición y orientación de la cabeza [15].

Los sistemas de *tracking* 3D son menos robustos que los que realizan seguimiento 2D, pero son mucho más precisos y permiten analizar el movimiento de todas las partes de la cara. Por el contrario, los sistemas de seguimiento 2D emplean primitivas muy simples por lo que son capaces de trabajar en peores condiciones.

Para este artículo hemos implementado un sistema de *tracking* 2D que detecta grupos de pixels de un color semejante al de la piel y está basado en el algoritmo de *Camshift* [2]. Este *traker* sigue el movimiento del mayor grupo de pixels que tengan este color, funcionando en dos pasos:

Paso 1:

Segmentación de la imagen adquirida en base al color de la piel, de forma que tendremos una serie de blobs indicando las zonas candidatas.

Paso 2:

Cálculo del área y del centro de gravedad de cada blob para focalizar el control sobre el de mayor tamaño. Tras la elección del blob de mayor tamaño, se dibuja un rectángulo para el marcado de la cara identificada.

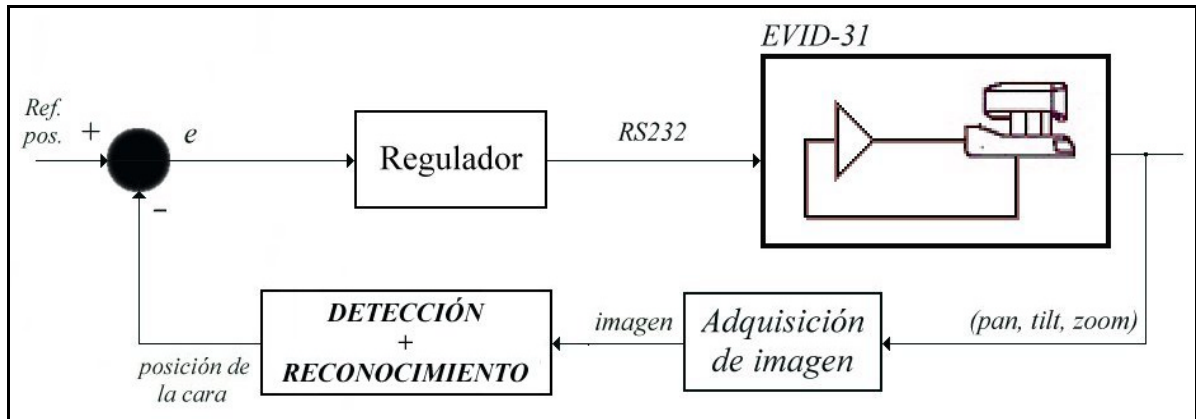


Figura 5. Esquema de control para el seguimiento de la cara basado en modelo.

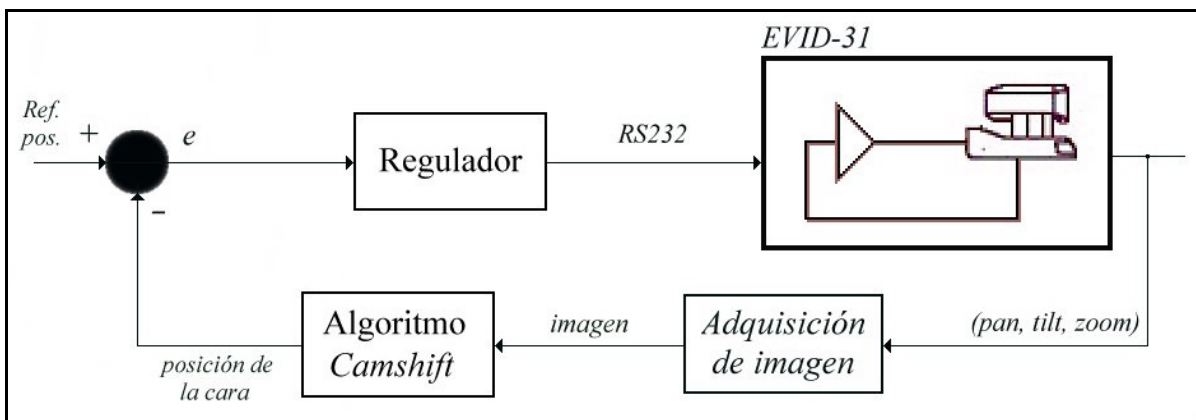


Figura 6. Esquema de control para el seguimiento de la cara basado en características.

3.2.2. Segmentación del color

Según algunos estudios [21], la distribución de color bajo distintas condiciones del entorno (p.e. iluminación) dependen más de la intensidad que del color en si mismo, por lo que normalizamos el modelo RGB respecto a la intensidad (R+G+B) obteniendo el modelo RG normalizado:

$$r_n = \frac{R}{(R + G + B)}, g_n = \frac{G}{(R + G + B)} \quad (1)$$

Con esta operación lo que hemos hecho es un mapeo de $\mathcal{R}^3 \rightarrow \mathcal{R}^2$.

La diferencia en la distribución del color se reduce tras la normalización. Es decir, el color piel de

diferentes personas es menos variable en el espacio de color normalizado (r_n, g_n) que en el (R, G, B).

Los pixels con color piel ocupan una determinada zona de (r_n, g_n) que puede modelizarse mediante una función $p(r_n, g_n | \alpha_p)$ que representa la probabilidad de que en la “clase piel” aparezca un píxel con color (r_n, g_n) . Por otra parte, la probabilidad de que ese píxel pertenezca a “clase no piel” $p(r_n, g_n | \alpha_{np})$, será una función uniforme ocupando todo el espacio de color, puesto que todo color que no sea piel tendrá la misma probabilidad de ser un color cualquiera.

Las funciones de probabilidad $p(r_n, g_n | \alpha_p)$ y $p(r_n, g_n | \alpha_{np})$ utilizadas son las propuestas por Yang [21].

En la Figura 7 podemos ver la normalización del color piel, mientras que en la Figura 8 tenemos el resultado de aplicar esta técnica sobre una imagen.

El resultado de la aplicación de estas funciones de probabilidad es una imagen en niveles de gris en la que cada píxel representa la pertenencia a la “clase piel”. En la Figura 8 aparece de color blanco las zonas que pertenecen totalmente a la “clase piel”, y en negro las que no pertenecen.

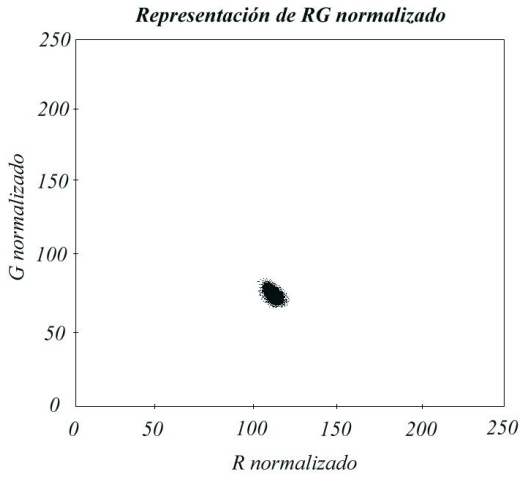


Figura 7. Espacio de color de los pixels de la cara



Figura 8. Resultado del proceso de segmentación

3.2.3. Seguimiento

El seguimiento se realiza a partir de los blobs obtenidos del procesamiento de las funciones de probabilidad y usando el algoritmo de *Camshift*, con la salvedad de que en la mayoría de las aplicaciones existentes se utiliza la técnica teniendo la cámara fija y en nuestro caso la cámara estará en movimiento para que la cara permanezca centrada en la imagen.

El esquema de control utilizado es el que podemos ver en la Figura 6, que ya habíamos referenciado.

Se ha implementado un regulador PID para resolver el control y se ha usado un ajuste experimental debido a la complejidad existente para la obtención

de una función de transferencia que nos permita un cálculo analítico.

La ecuación en diferencias del PID empleado tanto en el regulador representado en la Figura 5 como en la 6 responde a la siguiente expresión:

$$u_k = u_{k-1} \cdot r_1 - u_{k-2} \cdot r_2 + e_k \cdot q_0 + e_{k-1} \cdot q_1 + e_{k-2} \cdot q_2 \quad (2)$$

donde:

$$q_0 = K_p \cdot (1 + \beta) \quad (3)$$

$$q_1 = -K_p \cdot (1 + \gamma - \alpha + 2\beta) \quad (4)$$

$$q_2 = K_p \cdot (\gamma - \alpha\gamma + \beta) \quad (5)$$

$$r_1 = \gamma + 1 \quad (6)$$

$$r_2 = \gamma \quad (7)$$

$$\alpha = \frac{T}{T_i} \quad (8)$$

$$\beta = N\gamma \quad (9)$$

$$\gamma = \frac{T_d}{NT + T_d} \quad (10)$$

y:

K_p es la constante proporcional

T_i es el tiempo de integral

T_d es el tiempo derivativo

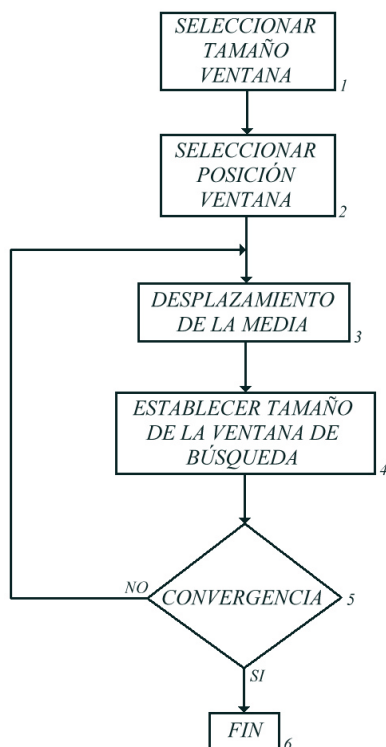
T es el periodo de muestreo

Los resultados obtenidos con la cámara permiten un seguimiento sin perder la cara del campo de visión siempre y cuando la cámara no tenga que realizar giros superiores a 22°/seg en el caso de seguimiento basado en características y de 13°/seg para el caso de seguimiento basado en modelo (detección y reconocimiento de la cara).

3.2.4. Algoritmo de *Camshift*

Este algoritmo se basa en estadística robusta, es decir, se tiende a desechar los datos atípicos de la muestra o los que se alejan demasiado del grupo, con lo que se tiende a eliminar el ruido.

El algoritmo trabaja como se muestra en el diagrama que podemos ver a continuación:



- 1.- Elección del tamaño de la ventana de búsqueda.
- 2.- Elección de la posición inicial de la ventana de búsqueda.
- 3.- Desplazamiento de la media (una o varias veces) y actualización del momento de orden 0.

El algoritmo de desplazamiento de la media se aplica a una imagen en escala de grises que indica la probabilidad de que cada píxel pertenezca a la clase piel (como ya habíamos planteado).

Donde el momento de orden 0 es:

$$M_0 = \sum_i \sum_j I(i, j) \quad (11)$$

donde $I(i, j)$ es el valor (probabilidad) del píxel en la posición (i, j) de la imagen, e (i, j) varían sobre toda la ventana de búsqueda.

- 4.- Elección del tamaño de la ventana de búsqueda igual a una función del momento de orden 0 encontrado en el paso 3.
- 5.- Evaluación de la convergencia. Si se cumple se finaliza el algoritmo, si no, se regresa al paso 3.

En otras palabras, el algoritmo de *Camshift* trabaja con una imagen de la distribución de la probabilidad del color, de forma que calcula el centroide de la distribución de probabilidad del color dentro de la ventana de búsqueda, vuelve a centrar la ventana y calcula el área para obtener el

siguiente tamaño de ventana. De esta forma no necesitamos volver a calcular la distribución de probabilidad del color en la imagen completa, sino que basta con restringir el cálculo de la distribución a una pequeña región de la imagen que envuelve a la ventana de búsqueda actual de *Camshift*.

Conclusiones

Basándose en el aprendizaje de la apariencia cromática de las caras humanas por un perceptrón, se ha desarrollado un detector de caras humanas que permite la segmentación de un escena proporcionando las posibles caras a reconocer. También se han realizado numerosos experimentos con métodos de reconocimiento de caras basados en proyecciones, destacando el buen funcionamiento del algoritmo PCA1 junto al clasificador MLP. Además, se han mostrado dos técnicas de seguimiento para caras humanas, una basada en el algoritmo de *Camshift* y otra que utiliza las etapas de detección y reconocimiento explicadas en apartados anteriores. En conclusión, se muestra un esquema sencillo de detección, reconocimiento y seguimiento de caras basado en la detección del color, su aprendizaje por parte de una red neuronal y el uso de eigenimágenes para el reconocimiento.

Agradecimientos

El trabajo realizado en este artículo se ha realizado gracias a la financiación aportada por la Generalitat Valenciana dentro del programa “Ayudas para proyectos I+D dirigidos a jóvenes investigadores , 2002-03” a través del proyecto CTDIA/2002/03.

Apéndice: Espacios o Modelos de color

Físicamente, la luz es parte de espectro electromagnético, y el color tiene que ver con la longitud de onda dentro del espectro visible (400 – 700 nm). La luz blanca consiste en la combinación de todos los colores de dicho espectro, entonces un objeto se ve de cierto color bajo una luz “blanca”, si refleja una luz de longitudes de onda alrededor de dicho color (ej. verde:800-570 nm) y absorbe el resto de longitudes de onda

La identificación de la información cromática de una imagen se hace mediante la combinación de tres señales, a estos se les denomina colores primarios (rojo, verde y azul). Mediante combinaciones de estos colores primarios es posible obtener toda la gama cromática, por lo que definen un sistema de coordenadas, el RGB. En el modelo cromático RGB los colores se representan en coordenadas cartesianas dentro de un cubo unitario. A partir del modelo RGB es posible definir otros modelos cromáticos lineales y no lineales. En la

Tabla 4 se muestran los modelos lineales usados en los experimentos de detección de caras.

| Espacio de color | Obtención de las coordenadas |
|------------------|--|
| YES | $Y = 0.253R + 0.648G + 0.063B$ $E = 0.500R - 0.500G + 0.000B$ $S = 0.250R + 0.250G - 0.500B$ |
| YIQ | $Y = 0.30R + 0.59G + 0.11B$ $I = 0.60R - 0.28G + 0.31B$ $Q = 0.21R + 0.52G + 0.31B$ |
| YUV | $Y = 0.299R + 0.587G + 0.114B$ $U = -0.147R - 0.289G + 0.437B$ $V = 0.615R + 0.515G - 0.100B$ |
| YCbCr | $Y = c_1R + c_2G + c_3B$ $Cb = (B - Y)/(2 - 2c_3)$ $Cr = (R - Y)/(2 - 2c_1)$ $c_1 = 0.2989^4$ $c_2 = 0.5866$ $c_3 = 0.1145$ |

Tabla 4: Espacios de color obtenidos mediante una transformación lineal del RGB

Referencias

- [1] Bichsel M. and Pentland A., "Topological Matching for Human Face Recognition", M.I.T. Media Lab, Vision and Modeling Group, Technical Report, No. 186, Jan. 1992.
- [2] G. R. Bradski. Computer vision face tracking for use in a perceptual user interface. Proceedings IEEE Workshop on applications of computer vision (WACV98), 214-219. 1998.
- [3] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(10):1042-1052, October 1993.
- [4] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," Proc. IEEE, vol. 83, pp. 705-740, May 1995.
- [5] Common, P., Independent component analysis, A new concept?, *Signal Processing*, 36:287--314, 1994.
- [6] Darrell T., Moghaddam B. and Pentland A., Active Face Tracking and Pose Estimation in an Interactive Room. *IEEE Conf. on Computer Vision & Pattern Recognition*, San Francisco, CA, June 1996.
- [7] M.T. Hagan and M.B. Menhaj. Training Feedforward Networks with the Marquardt Algorithm. *IEE Trans on Neurak Networks*, 5(6):989-993, Nov. 1994.
- [8] G. Hager, P.N. Belhumeur. Real-time tracking of image regions with changes in geometry and illumination. Proceedings International Conference on Computer Vision and Pattern Recognition, 403-410. 1996.
- [9] E. Hjelmás and B.K. Low, "Face detection: A survey," *Journal of Computer Vision and Image Understanding*, vol. 83, pp. 236-274, 2001.
- [10] R. L. Hsu, M. A. Mottaleb, and A. K. Jain, "Face detection in color images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696-706, May 2002.
- [11] Aapo Hyvärinen, Juha Karhunen, Erkki Oja, "Independent Component Analysis", Wiley Interscience, 2001.
- [12] M. Kirby and L. Sirovich. *Application of the Karhunen-Loeve procedure for the characterization of human faces*. *IEEE PAMI*, 12(1):103--108, 1990.
- [13] Moghaddam B., Wahid W. and Pentland A., Beyond Eigenfaces: Probabilistic Matching for Face Recognition *International Conference on Automatic Face & Gesture Recognition*, Nara, Japan, April 1998.
- [14] Pentland A., Moghaddam B., Starner T., View-Based and Modular Eigenspaces for Face Recognition *IEEE Conf. on Computer Vision & Pattern Recognition*, Seattle, WA, July 1994.
- [15] R. Stiefelhagen, J. Yang, A. Waibel. A model-based gaze tracking system. *International Journal on Artificial Intelligence Tools*, 6(2) 193-209. 1997.
- [16] K.K.Sung, and T.Poggio, "Example-based learning for view-based human face detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20 (1),1998,pp.23-38.
- [17] Turk, M., Pentland, A., Eigenfaces for Recognition, *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp.71-86,1991.
- [18] M. A. Vicente, O. Reinoso, C. Fernández, L. Jiménez "Experiments with COIL database using PCA-based Object Recognition Techniques", Proceedings of the 3rd International Symposium on Robotics and Automation ISRA'2002.
- [19] Wechsler H. et al. *Face Recognition, From Theory to Applications*.
- [20] M. H. Yang, D. J. Kriegman, N. Ahuja, "Detecting faces in images: A survey," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34-58, Jan. 2002.
- [21] J. Yang, W. Lu, A. Waibel. Skin-color modeling and adaptation. Proceedings Third Asian Conference on Computer Vision, Vol. II, 142-147. 1998.
- [22] W..Zhao., R.Chellapa and P.J. Phillips. Subspace Linear Discriminant Analysis for Face Recognition.1999

⁴ según el standard Rec.601