



## **Detección de Rostros Humanos en Posición Frontal en Imágenes a Color Utilizando Propiedades Estadísticas de la Piel Humana Junto con el Método de Concordancia con el Rostro Plantilla**

Marlon Fabián Macías Sánchez<sup>(1)</sup>, Patricia X. Chávez Burbano<sup>(2)</sup>

Facultad de Ingeniería Eléctrica y Computación

Escuela Superior Politécnica del Litoral

Km 30.5 vía Perimetral, Código Postal: 09-01-5863, Guayaquil, Ecuador

[mfmacias@espol.edu.ec](mailto:mfmacias@espol.edu.ec) <sup>(1)</sup>, [pchavez@espol.edu.ec](mailto:pchavez@espol.edu.ec) <sup>(2)</sup>

### **Resumen**

*Un método para detectar rostros humanos en imágenes a color fue desarrollado e implementado, el cual esta conformado de tres pasos fundamentales: Estadísticas del color de la Piel Humana, Proceso Adaptativo de Umbral y el de Concordancia con la Plantilla. Un modelamiento de la piel humana determina las regiones más probables de que correspondan a una zona de piel dentro de una imagen. Este modelo fue elaborado a partir de 20 muestras de tonos de piel de personas de distinta raza, cuya distribución en el espacio de color cromático genera un chroma chart mostrando probabilidades de los colores de piel. Luego la imagen es transformada a escalas de grises usando este chart; donde el valor de un pixel dado muestra la probabilidad de que dicho pixel represente un pixel de piel. Esta imagen en escala de grises es segmentada a regiones de piel y de no piel usando un proceso adaptativo de umbral para obtener el valor optimo del umbral. Finalmente, la concordancia con el Rostro Plantilla determina si una región de piel dada representa un rostro humano.*

**Palabras Claves:** Chroma-Chart, Skin Color Regions, Face Detection, Template Matching.

### **Abstract**

*A method for detecting human faces in color images was developed and implemented, consisting of three main steps: human color statistics, adaptive thresholding process and Template Matching. A skin model determines the most likely skin regions in an image. This skin model is made from 20 skin samples from persons of different ethnicity, which color distribution in the chromatic color space generates a chroma chart showing likelihood of skin colors. The image is transformed into a gray scale image using this chart; the gray value at a pixel shows the likelihood of the pixel representing skin. Then this gray scale image is segmented to skin and non-skin regions using an adaptive thresholding process to achieve the optimal value. A fixed threshold value is not possible to be found, since people with different skins have different likelihood. Finally, the template matching determines if a given skin region represents a frontal human face.*

**Palabras Claves:** Chroma-Chart, Skin Color Regions, Face Detection, Template Matching.

## 1. Introducción

Un sistema enfocado en la Detección de Rostros humanos nos ayudaría a determinar la presencia y subsecuentemente la posición de un rostro o rostros humanos en una imagen, la cual es una tarea importante en el campo de Reconocimiento de Patrones, de hecho la Detección de rostros Humanos es el primer paso en un Sistema de Reconocimiento de Patrones. Puesto que los humanos son hábiles para detectar e identificar rostros en una escena con un pequeño o casi nulo esfuerzo, debido a que nuestro cerebro analiza, segmenta, clasifica, reconoce e interpreta una gran cantidad de datos capturados mediante los órganos sensoriales como por ejemplo los ojos (información visual), y los oídos (información auditiva) cuando miramos en una escena. Estas tareas son desarrolladas por nuestro cerebro dentro de fracciones de segundo con un bajo error.

El propósito de este proyecto es el de detectar rostros humanos en varias imágenes a color. Hay diferentes aplicaciones en las cuales un programa de detección de rostros podría ser usado. Una de las cuales podría ser, usar este sistema junto con una base de datos que contengan imágenes. Por ejemplo, si se quiere buscar a alguien en especial en una base de datos que contiene varias imágenes de rostros humanos, el usuario podría simplemente pedirle al programa que encuentre imágenes que contengan personas específicas, las cuales podría ser determinadas mediante la detección de rostros de personas en cada imagen para después que una vez que se hayan detectados proceder con el paso de Reconocimiento de Patrones. También, detección de rostros es el primer escalón en los procesos de Visión por Computadores, en el cual se intenta que un robot vaya adquiriendo ciertas características humanas para que ejecute tareas específicas. Hoy en día varias compañías que se dedican al peritaje de personas involucradas en actos delictivos podrían hacer uso de este tipo de programas, los cuales pueden confiablemente escanear en una escena a una persona considerada sospechosa, y así tomar acciones contra este.

Así para reconocer una persona en una imagen, primero es necesario encontrar el rostro de esa persona en la imagen. Este tipo de programas también podría ser de mucha utilidad en aeropuertos, por ejemplo si estos quieren encontrar criminales. De esta manera la necesidad de detección de rostros se ha vuelto más imperativa en las aéreas de seguridad y también en aplicaciones como teleconferencias, telecomunicaciones e interacción de humanos con computadoras.

## 2. Implementación del Sistema

El sistema de Detección de Rostros que se desarrollo usa MATLAB para localizar rostros humanos en una imagen dada, donde la salida de este sistema es la imagen de entrada pero con rectángulos ubicados sobre los rostros detectados. El primer paso en este programa es construir un modelo de la piel humana para determinar las regiones de piel más probables en la imagen. Este modelo de la piel fue construido a partir de una variedad de muestras de piel de personas de distintas razas. Luego lo utilizamos para crear una imagen en escala de grises donde la intensidad de cada pixel representa la probabilidad de que ese pixel represente un pixel correspondiente a una región de piel, luego esta nueva imagen será transformada a una imagen binaria en base a un valor optimo de umbral. Después se efectúan operaciones morfológicas junto con el proceso de concordancia con el rostro plantilla, para determinar si las regiones resultantes después de las operaciones morfológicas representan o no un rostro. Finalmente un rectángulo es ubicado sobre cada rostro detectado.

### 2.1. Modelamiento del Color de Piel

El primer paso para el desarrollo de este programa es modificar la imagen original, en la cual estamos interesados en detectar los rostros. Para eliminar los efectos de luminosidad, es necesario tomar la imagen a color de entrada al sistema y convertirla al espacio YCbCr, o comúnmente llamado Espacio de Color Cromático o de Colores Puros. Debido a que en el espacio en el cual se haya la imagen a color original, no es confiable para la caracterización de colores de piel, debido a que las tres componentes (r, g, b) representan no solo color, sino que también representan la luminosidad o luminancia, el cual puede variar el rostro de una persona debido a condiciones ambientales de luz, por ende no es una medida confiable en la separación de regiones que contienen piel de las que no contienen. Luminancia puede ser eliminada de la representación de color, llevándola al espacio de color cromático. Así para convertir una imagen del espacio RGB al espacio YCbCr, tenemos que simplemente que calcular:

$$r = R/(R + G + B) \quad (1)$$

$$b = B/(R + G + B) \quad (2)$$

Donde el valor de la componente g es de la misma forma que r y b, además  $r + b + g = 1$ . Cabe recalcar que para convertir una imagen desde el espacio RGB al espacio cromático en MATLAB, la función "rgb2ycbcr" es utilizada. Las siguientes imágenes

muestran a la imagen original que se encuentra en el espacio RGB, y en el espacio cromático.



**Figura 2.1** Espacio de Color RGB.

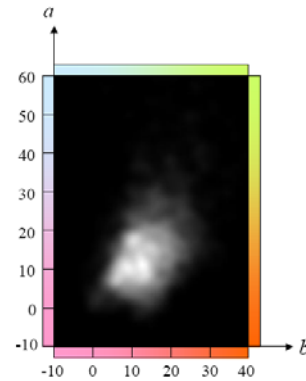


**Figura 2.2:** Espacio de Color YCbCr

En La distribución del color de la piel de personas de diferentes razas, se encontró que están agrupadas en una pequeña área del espacio de color cromático. Aunque, el color de piel de diferentes personas parece variar sobre un amplio rango, pues no es así, ellas difieren mucho menos en color que en brillo. En otras palabras el color de piel de diferentes personas está agrupado, pero difieren principalmente en intensidad [4].

El gráfico Chroma obtenido en [4] por Cai y Goshtasby es el resultado del procesamiento de 2300 muestras de piel obtenidas de un amplio rango de colores de piel provenientes de 80 imágenes de prueba y encontraron que todas ellas caen dentro de una pequeña región dentro del espacio CIE LAB 1976, a  $\in [-10, 60]$  y b  $\in [-10, 40]$ . Donde las dos componentes a y b representan los valores de a y b en el Espacio Coordinado CIE LAB 1976.

El gráfico cromático de la Figura 2.3 fue obtenido de 2,300 muestras de piel, tomadas de un amplio rango de colores de piel en 80 imágenes. El eje horizontal muestra la componente a y el eje vertical muestra la componente b del gráfico cromático.



**Figura 2.3** Chroma Chart.

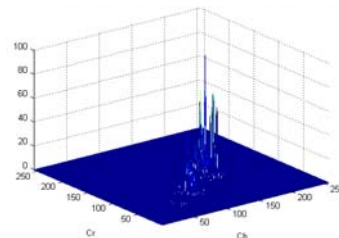
El siguiente paso es crear nuestro modelo de la piel en el espacio de color cromático. Para crear el modelo de la piel, fue necesario usar algunas imágenes con personas de variada tonalidad de piel. Yo use 20 muestras de 16x16 pixeles, que fueron recortadas de las imágenes de prueba que utilice en mi proyecto, cabe recalcar que las muestras contienen solamente regiones que corresponden a piel humana. Después de leer en MATLAB cada una de las muestras de piel, ellas fueron convertidas al espacio de color cromático como previamente mencione. Cada unas de estas muestras fue entonces filtrada por un filtro pasa bajo para así remover el ruido en cada imagen.

La respuesta al impulso del filtro pasa bajo utilizada es la siguiente:

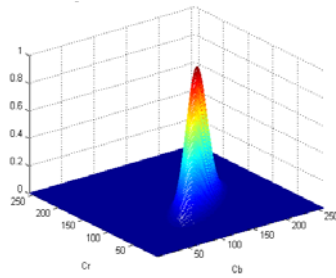
$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

**Figura 2.4** Respuesta al Impulso del Filtro Pasabajo.

Luego el histograma del color reveló que la distribución del color de piel de las diferentes personas se encuentran agrupada en el Espacio de color cromático y que el color de piel puede ser representada por un modelo Gaussiano  $N(m, c)$ .



**Figura 2.5** Histograma del Color de Piel.



**Figura 2.6** Distribución Gaussiana Color de Piel.

Donde tenemos que:

Media:  $m = E \{x\}$  donde  $x = (r \ b)^T$  (3)

Covarianza:  $C = E \{(x - m)(x - m)^T\}$  (4)

Así usando la media y la covarianza, el modelamiento del color de piel puede ser encajado en un modelo Gaussiano, como el mostrado en la Figura 2.6, donde se muestra la componente  $r$  vs la componente  $b$ , ambas del Espacio de Color Cromático.

Ahora que tenemos un modelo para la región de piel nos toca compararlo con nuestra imagen de entrada, y así de esta manera podemos comenzar el proceso de detección de rostros. El siguiente paso es determinar regiones en la imagen original que son más probable de ser una región de piel. Con el modelo de Distribución de la piel mostrado en Figura 2.6, se procede a determinar regiones que son probables de ser piel. Por consiguiente, si un pixel, habiendo sido transformado desde el espacio de color RGB al Espacio de Color Cromático, tiene una pareja de valores cromáticos de  $(r, b)$ , la probabilidad de que este pixel pueda representar un pixel de piel puede entonces ser calculada como sigue:

$$\text{Probabilidad} = P(r, b) = \exp [-0.5 * (x - m)^T C^{-1} (x - m)] \text{ where } x = (r \ b)^T \quad (5)$$

De esa manera, el modelo del color de la piel puede transformar una imagen a color en una imagen de escalas de grises tal que la intensidad en cada pixel muestra la probabilidad de que el pixel pertenezca a una región de piel. Con un apropiado valor de umbral, la imagen en escala de grises puede entonces ser transformada a una imagen binaria, mostrando de esta manera las regiones que corresponden a piel y aquellas que no corresponden.

Las siguientes imágenes muestran la imagen original junto a la imagen en escala de grises cuyos pixeles representan la probabilidad de un pixel de pertenecer a una región de piel.



**Figura 2.7** Imagen de Probabilidades.

## 2.2. Proceso Adaptivo Para Encontrar Umbral Óptimo

Con este modelo Gaussiano del color de piel, obtuvimos un gráfico chroma que muestra la manera en que se encuentran agrupados los distintos colores de piel humana. Este gráfico chroma se usó para generar una imagen en escala de grises de la imagen a color original. Esta imagen tiene la propiedad que el valor gris en un pixel cualquiera, muestra la probabilidad de que ese pixel represente pertenezca a una muestra de piel.

Aunque es importante notar que la detección de regiones puede incluir regiones que necesariamente no pertenezcan a piel. Por eso solamente cabe concluir que esas regiones detectadas tienen el mismo color como que el de la piel humana. El punto importante aquí es que este proceso puede confiablemente sacar regiones que no necesitan ser consideradas jamás en el proceso de detección de rostros.

Debido que las regiones de piel son más brillantes que otras partes en la imagen de escala de grises, las regiones de piel pueden ser segmentadas del resto de la imagen a través de un proceso de Segmentación, donde el valor del umbral para realizar la segmentación no será un valor fijo para todas las imágenes de entrada puesto que en las imágenes hay personas que poseen diferentes tipos de piel, por lo tanto un valor fijo de umbral no es posible. Puesto que personas con diferentes tonos piel, tienen diferentes probabilidades, un Proceso Adaptivo de Segmentación es requerido para lograr encontrar el valor de umbral óptimo.

El Proceso Adaptivo de Segmentación está basado en la observación de que si disminuimos el valor del umbral, es decir si el valor del umbral es demasiado bajo, la cantidad de región segmentada se incrementa. Basado en el hecho que cuando el valor del umbral disminuye y es demasiado bajo, el incremento en el número de regiones de piel se dispara, causando que otras regiones que no corresponden a piel sean



incluidas. El valor del umbral al cual el mínimo incremento en el tamaño de la región de piel es observado mientras disminuimos el valor de umbral será el valor óptimo. En mi programa, el valor de umbral es disminuido desde 0.65 hasta 0.05 en pasos de 0.05. Por ejemplo si el mínimo incremento en la región ocurre cuando el valor de umbral fue cambiado desde 0.45 a 0.35, entonces el valor de umbral óptimo será tomado como 0.4. Usando esta técnica de Segmentación Adaptiva, varias de las imágenes obtendrán buenos resultados al momento de eliminar aquellas regiones que no corresponden a regiones de piel; de esta manera las regiones que corresponden a piel son efectivamente segmentadas de las regiones que no corresponden a regiones de piel.

Una vez que el valor de umbral óptimo es determinado, la imagen en escala de grises es convertida a una imagen binaria.

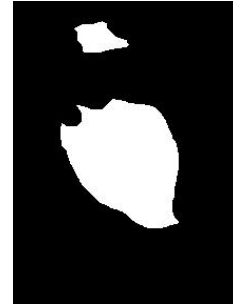


**Figura 2.8** Imagen Segmentada o Binaria.

### 2.3. Operaciones Morfológicas

Ahora que tengo la imagen binaria con 1's representando pixeles de piel y 0's representando pixeles de no piel. Entonces hay la necesidad de aplicar las operaciones morfológicas tales como relleno, erosión y dilatación para separar áreas de piel que está conectadas a regiones de no piel que lograron sobrevivir a la segmentación. Primeramente relleno los huecos en la imagen binaria, para la cual utilizo la función "imfill" de MATLAB. Luego una erosión es aplicada para reducir aquellas áreas, junto con una dilatación para hacer crecer las áreas de piel las cuales fueron pérdidas debido a la agresiva erosión aplicada en el paso previo, en MATLAB tenemos "imerode" y "imdilate", las cuales ejecutan los procesos de erosión y dilatación respectivamente. Finalmente la imagen binaria dilatada es multiplicada con la imagen binaria producto del proceso de segmentación para mantener los huecos. Esto es debido a que los utilizare en el siguiente paso, el cual se basa en el número de huecos para eliminar regiones que no corresponden a regiones de rostros.

En las figuras 2.9 y 2.10 se muestran un ejemplo de cómo quedan las imágenes después del proceso de erosión y del proceso de dilatación.



**Figura 2.9** Imagen Erosionada.



**Figura 2.10** Imagen Dilatada.

### 2.4. Etiquetado de Regiones

La imagen binaria resultado después de las operaciones morfológicas necesita ser etiquetada para que cada grupo de pixel pueda ser clasificado como una región, y así cada región pueda ser analizada más adelante para determinar si esta corresponde a un rostro o no. Esto es en vez de tener 1 y 0, cada región es etiquetada como 1, 2, 3 y así sucesivamente, dependiendo del número de regiones que se encuentren. Los pixeles con valor 0 no son cambiados, permanecen iguales.

El proceso de determinar cuántas regiones tenemos en una imagen binaria es mediante la puesta de etiquetas de tales regiones. Una etiqueta es un valor entero. I use los 8 pixeles vecinos de un determinado pixel para proceder a etiquetar las regiones. Si cualquiera de los vecinos de un pixel tiene una etiqueta, nosotros etiquetamos el pixel actual con esa etiqueta, si no, entonces usamos una nueva etiqueta. Al final cuento el número de etiquetas y esta será el número de regiones que hay en la imagen.

En MATLAB la función bwlabel me permite realizar el proceso de etiquetación.



**Figura 2.11** Regiones Etiquetadas.

### 2.5. Distancia de Euler

En esta etapa cada región resultado del paso anterior es procesada a la vez. El objetivo de esta etapa es eliminar regiones que no contienen huecos, esto es debido a la asunción de que un rostro humano contiene al menos un hueco, puesto que los rostros humanos consisten de ojos, narices y boca, entonces sería seguro decir que un rostro contiene al menos un hueco en su rostro. Para determinar el número de huecos dentro de una región, necesitamos calcular el número de Euler de la región. Este se define de esta manera:

$$E = H - C \quad (6)$$

Donde E: es el número de Euler, C: el número de componentes conectadas y H: el número de huecos en una región.

Puesto que estamos analizando solo una región segmentada a la vez, el valor de C es igual a 1, entonces tenemos que:

$$H = 1 - E \quad (7)$$

El número de Euler es determinado mediante la función "bweuler" en MATLAB.

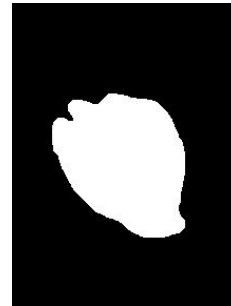


**Figura 2.12** Imagen después del Test de Euler.

### 2.6. Relación Ancho Alto

El ancho y el alto de la región son usados para mejorar el proceso de decisión. A fin de obtener menores pérdidas en las regiones, se determinó experimentalmente que un valor de relación ancho alto mínimo de 1 funciona bien. Valores por debajo de 1 no nos sugieren un rostro humano orientado de manera frontal.

Así como se tiene un mínimo esta relación debería también tener su valor máximo, que fue determinado mediante el análisis de resultados en nuestro experimento, el cual sugiere que funcionaría bastante bien un valor de 3.5. Así cualquier región con una relación de aspecto improbable de ser un rostro es descartada.



**Figura 2.13** Imagen después del Test de Aspecto.

### 2.7 Concordancia con la Plantilla

El proceso de Concordancia con la Plantilla toma el área segmentada, y coloca un rostro promedio o un rostro plantilla sobre esta región. El rostro plantilla es cambiado de tamaño y posicionado de acuerdo a las propiedades de la región segmentada que fueron tomadas tempranamente en el programa. Este programa de Detección de Rostro Humanos usa una plantilla para determinar si la región segmentada que tiene al menos un hueco dentro de ella pertenece o no a un rostro humano. El rostro plantilla utilizado aquí ha sido del resultado de un promedio de 16 diferentes rostros humanos tomados de hombres y mujeres que no usaban lentes y que no tenían bellos faciales. Estas imágenes fueron seleccionadas de las base de datos de las Universidad de Yale y de la Universidad de Bern. El rostro plantilla primero se ajusta en tamaño de acuerdo a las medidas de la imagen segmentada. Basadas en la relación alto para el ancho de la región de piel, el rostro plantilla es entonces convertido a esas dimensiones de tal manera que pueda después ser puesto en la región segmentada.



**Figura 2.14** Rostro Plantilla.

El ángulo de la región segmentada es usado para rotar el rostro plantilla al mismo ángulo. El centro de masa de la región segmentada es usado para ubicar el rostro plantilla directamente en el centro de la imagen segmentada. Este proceso llenara el área segmentada con la imagen del rostro plantilla.

Una vez que el rostro plantilla es puesto dentro de la imagen segmentada, es necesario determinar cuán bien encaja la plantilla dentro de la región. Una manera de hacer esto de calcular la correlación, la cual calcula el coeficiente de correlación bidimensional entre dos matrices. Para encontrar la correlación de estas dos matrices, usamos la función “corr2” de MATLAB. La cual se basa en la operación del siguiente algoritmo.

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right) \left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}}$$

Probando varios números, se encontró que un buen valor de correlación entre las dos matrices debe ser muy cercano a 0.6. Si la correlación entre la matriz de la imagen plantilla con la matriz de la región segmentada es 0.6 o mayor, la imagen se muestra con el rostro plantilla reemplazando esta región y es mostrada como una imagen en escala de grises.

El mismo proceso es repetido con todas las regiones. Una vez que todas las regiones de la imagen han sido evaluadas, el producto final es la imagen a color de entrada con rectángulos encerrando cada rostro detectado en la imagen.



**Figura 2.15** Concordancia del Rostro Plantilla.



**Figura 2.16** Imagen Resultante.

### 3. Conclusión y resultados.

Con las imágenes resultados y analizando los resultados expuestos en la Tabla I, puedo expresar lo siguiente:

Si nos ubicamos en la última columna de la Tabla I, podemos darnos cuenta que a medida que el número de personas se van incrementando en las imágenes, el algoritmo se demora cada vez más en ejecutar el proceso de Detección de Rostros.

Si queremos saber la eficiencia del algoritmo, es decir cuán preciso es para detectar rostros los único que tenemos que hacer es hacer una relación entre el número de rostros que detecta para el número de rostros presentes en las imágenes, con lo cual obtenemos que la eficiencia para este programa es de casi un 89%, pero con un porcentaje de Falsos Positivos del 16% y un porcentaje de Falsos Negativos del 12%.

El Algoritmo funciona bastante bien para imágenes que contienen un solo rostro donde el fondo de la imagen sea sólido, es decir que no contenga objetos, que pueden dar la oportunidad a que aparezcan Falsos Positivos.

A medida que el número de rostros aumenta en la imagen el error en la detección de rostros también lo hace, esto es debido a que el tamaño de los rostros se vuelve pequeño, y las características faciales dentro de los rostros son frecuentemente perdidos.

Hay situaciones en la que se seleccionan objetos que están en la imagen, pero que no representan rostros, esto es debido a que algunos de estos tienen una similitud a la del color de la piel humana.

Las pérdidas de rostros en el presente algoritmo fueron debido a las siguientes razones: (1) Los rostros fueron demasiados pequeños, o fueron separados en pequeñas piezas, que las características faciales son completamente perdidas, haciendo que estas regiones no sean contempladas como rostros. (2) Dos rostros estuvieron tan cerca uno del otro lo que hace parecer

que fueran una sola región. (3) hay rostros que no están en una posición frontal.

**Tabla I** Resultado del Algoritmo de Detección de Rostro.

Imagen	Rostros en Imagen	Rostros Detectados	Falsos Positivos	Falsos Negativos	CPU Time (s)
Ensayo 1	1	1	0	0	10.45
Ensayo 2	1	1	0	0	12.14
Ensayo 3	1	1	0	0	6.51
Ensayo 4	1	1	0	0	6.77
Ensayo 5	1	1	0	0	6.35
Ensayo 6	1	1	0	0	8.41
Ensayo 7	2	2	0	0	8.98
Ensayo 8	3	3	0	0	9.99
Ensayo 9	3	3	1	0	20.55
Ensayo 10	4	2	0	2	22.11
Ensayo 11	4	4	1	0	24.16
Ensayo 12	14	10	1	4	31.16
Ensayo 13	8	6	1	2	26.71
Ensayo 14	6	5	0	1	21.13
Ensayo 15	23	22	0	1	37.60
Ensayo 16	21	19	4	3	49.79
Ensayo 17	15	15	9	0	49.10
<b>Total</b>	<b>109</b>	<b>97</b>	<b>17</b>	<b>13</b>	

- [3] C. Garcia and G. Tziritas, "Face Detection Using Quantized Skin Color Regions Merging and Wavelet Packet Analysis", IEEE transactions on multimedia, vol. 1, No. 3, pp. 264-277, Sept 1999.
- [4] J. Cai & A. Goshtasby & C. Yu, "Detecting Human Faces in Color Images", Wright State University, U. of Illinois.
- [5] Jie Yang and Alex Waibel, "A Real-Time Face Tracker", CMU CS Technical Report.
- [6] <http://www.syseng.anu.edu.au/~luke/cvcourse.htm>
- [7] EE368 Lecture Notes, Professor Bernd Girod, Stanford University.
- [8] Página Web de Detección de Rostro, esta es la URL:
- [9] <http://home.t-online.de/home/Robert.Frischholz/index.html>

## 4. Agradecimientos

De manera especial, queremos extender nuestro agradecimiento al Dr. Boris Ramos por brindarnos su ayuda e importantes comentarios respecto al presente estudio.

## 5. Referencias

- [1] H.A. Rowley, S. Baluja, and T. Kanade, "Neural Network-Based Face Detection", IEEE Trans. Pattern Anal. Machine Intell., vol. 20, pp. 23-28, 1998.
- [2] O. Jesorsky, K. J. Kirchberg, and R.W. Frischholz, "Robust Face Detection Using the Hausdorff Distance", Third International Conference on Audio- and Video-based Biometric Person Authentication, Springer, Lecture Notes in Computer Science, LNCS-2091, pp. 90-95, Halmstad, Sweden, Jun 2001.