

Universidad de las Ciencias Informáticas



Facultad#6

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

*Algoritmo para la detección de escenas que contienen rostros de personas
en secuencias de videos de noticias.*

Diplomante:

Aliuska Castañeda Martínez

Tutor:

Ing. Magdiel Rivero González

Ciudad de la Habana, Cuba.

2011

Yo: **Aliuska Castañeda Martínez** me declaro como único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas (UCI) para que haga uso de este trabajo de la manera que estime conveniente.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____

Aliuska Castañeda Martínez

Magdiel Rivero González

Firma del Autor

Firma del Tutor



“Creo en el milagro de lo que puede hacer el trabajo, de lo que puede hacer la ciencia, de lo que pueden hacer los hombres”.

Fidel Castro Ruz

....A Dios por permitirme cumplir mi sueño y darme las fuerzas necesarias para seguir adelante en todo momento.

.... A mi papito por apoyarme, por ser mi fuente de inspiración, esencia de todo cuanto soy en la vida, el motor impulsor de este sueño, Papi, te agradezco que hayas sido tan duro conmigo, porque me enseñaste que en la vida hay que luchar para conseguir lo que uno quiere. Te quiero mucho papi.

.....A mis abuelos los seres más comprensibles y hermosos que tengo en este mundo, por estar siempre a mi lado y no cansarse de darme consejos.

.... A mi mamá por apoyarme de forma incondicional durante toda la carrera y darme fuerzas para seguir adelante.

..... A mis tíos y primos por su apoyo, su ayuda desinteresada y por la confianza que depositaron en mí, les estaré eternamente agradecida.

..... A mi hermana Yiset mi tesoro más preciado, te adoro....

.... A mi cosi, por demostrarme que la vida sí vale la pena y que se puede llegar muy lejos.

... A Daya y Vladi por ser mis segundos padres y estar siempre que los he necesitado.

....A Raidel, quien me acompañó durante 4 años de esta carrera, a ti que siempre me decías que lo primero era mi escuela y después nuestra relación.

... A mis amigas Lisette, Melixa y Yisel ustedes que han compartido conmigo trabajo, amor, sonrisas, consejos, tristezas y momentos inolvidables. Las quiero.

....A mis amigos Jose, Gatico, Yasiel, Daniel, Dixan, por hacer que estos 5 años fueran inolvidables.

... A mis compañeros de aula Ivi, Yara, Siome, Eli, Alain, Ivelin, Lisandra, Lusmey, nunca los olvidaré.

.... A mis amigos de fiestas Alleyne, Sandy, Osmel, Jeem, los reyes de la UCI.

... A mi tutor Magdiel por hacer su mayor esfuerzo en ayudarme.

... A mis amigos del 9108....

... A mi perrito Jhony.....

... A Todas las personas que me han ayudado y que no puse, pero no por no acordarme de ellos, sino por el espacio.

...gracias a todos.....

Dedico este trabajo de diploma a:

A mis padres, en especial a ti papi, te debo todo lo que soy.

A mis abuelos Magda y Wilson, tienen una nieta ingeniera.

A mi hermana Yiset, espero ser un ejemplo para ti.

A mis amigas Melixa, Lisette y Yisel, las quiero mucho.

A mi cosi, por ser lo más lindo que me sucedió en estos 5 años

A toda mi familia por estar siempre presente cuando los necesito y nunca me han dado la espalda.

A todos mis amigos que de una forma u otra me han brindado su apoyo cuando más los necesité.

A la Revolución a Fidel y a la UCI.

La detección de rostros no es más que determinar en una imagen o video la presencia de uno o más rostros y si es el caso, devolver su posición y escala. Esta resulta ser una fase elemental en cualquier aplicación donde se realice algún tipo de análisis facial como por ejemplo reconocimiento de rostros, codificación de video en videoconferencias, interfaces inteligentes hombre-máquina, entre otros.

El objetivo de esta tesis es proponer un algoritmo capaz de detectar las escenas en las que aparecen caras de personas en secuencias de video noticias. Por lo que se realiza un estudio detallado sobre los algoritmos existentes así como las herramientas a utilizar, con el fin de facilitar la comprensión de la importancia de la investigación, su alcance y aporte científico. Presentando como solución, el algoritmo de detección de rostros Adaboost. Además para probar su correcto funcionamiento se implementa un prototipo funcional al cual se le realizan pruebas de rendimiento donde queda demostrado el correcto funcionamiento de la propuesta de solución.

Face detection is nothing more than determine in an image or video the presence of one or more faces and if so, return its position and scale. This turns to be an essential phase in any application able to make a face analysis, for example face detection, video coding in video conferences, intelligent interfaces man – machine, among others.

The objective of this work is to propose an algorithm able to detect scenes in which appear people faces in news video sequences. Thus, a detailed study is made on existing algorithms as well as on the tools to use, aiming at facilitating the comprehension of the importance of the research, its approach and scientific contribution. Presenting as solution, the face detection algorithm Adaboost. Apart from that, to prove its correct functioning, it is implemented a functional prototype which is tested for performance demonstrating the correct functioning of the proposed solution.

Índice de contenidos

Introducción.....	1
Capítulo1: Fundamentación Teórica.....	4
Introducción	4
1.1 Conceptos asociados al dominio del problema.....	4
1.2 Objeto de Estudio.....	6
1.3 Descripción General.....	6
1.4 Soluciones Existentes	8
1.4.1. Detección de rostros	8
1.4.1.1. Segmentación.....	8
1.4.1.2. Áreas potenciales.....	10
1.4.1.3. AdaBoost	11
1.4.1.4. Detección utilizando elipses.....	13
1.4.1.5. Redes Neuronales Artificiales para la detección de rostros.....	14
1.5 Elementos computacionales.....	15
1.6 Conclusiones	18
Capítulo 2: Presentación de la Solución Propuesta	19
Introducción.....	19
2.1. Conceptos Fundamentales	19
2.2. Detección Geométrica.....	19
2.3. Propuesta de métodos para la detección de rostros.....	25
2.4. Conclusiones	27
Capítulo 3: Pruebas al algoritmo propuesto.	28
Introducción.....	28
3.1 Prototipo Funcional	31
3.2 Conclusiones	33
Conclusiones Generales	34
Recomendaciones.....	35
Bibliografía	36

Índice de Contenidos

Referencias Bibliográficas	38
Glosario de Términos.....	39

Índice de figura

Figura 1. Imagen digital.	5
Figura 2. Pixel.....	5
Figura 3. Video digital.	6
Figura 4.Detección de rostros	7
Figura 5. Características de Haar-like.....	12
Figura 6. Triángulo de verificación	24
Figura 7. Combinaciones de métodos.....	25
Figura 8 Videos a los que se les realizaron las pruebas	28
Figura 9 Falsos Positivos.....	29
Figura 10 Falsos Positivos.....	29
Figura 11 Efectividad del algoritmo luego de aplicada la segmentación.....	30
Figura 12 Efectividad del algoritmo luego de aplicada la segmentación.....	31
Figura 13 Detección de escenas que contienen rostros.....	31
Figura 14 Buscando ubicación del video.....	32
Figura 15 Detección de rostros.	32
Figura 16 Secuencias detectadas.....	33

Índice de Tablas

Tabla 1. Leyenda	26
Tabla 2. Caminos.....	26
Tabla 3 Resultados de Efectividad	29
Tabla 4 Totales	30

Introducción

Internet, conocida como la red de redes, se ha convertido en el medio de comunicación por excelencia en la actualidad. Gracias a ella las personas en todo el mundo mantienen una estrecha comunicación y disfrutan de múltiples servicios. Al producirse la aparición de la Internet comercial, las empresas se ubican en la red y se ofrecen todo tipo de servicios online. Dentro de esta ola de digitalización se vieron inmersos también los audiovisuales y su transmisión.

La transmisión de video sobre redes de telecomunicaciones está llegando al punto de convertirse en un sistema habitual de comunicación debido al crecimiento masivo que ha supuesto Internet en estos últimos años. Se está utilizando para ver películas o comunicarse con conocidos, pero también se usa para dar clases remotas, para hacer diagnósticos en medicina, videoconferencia, video bajo demanda, para distribuir multimedia en Internet y televisión digital.

La información que se brinda a través de los medios audiovisuales ocupa una parte importante de la programación en las cadenas públicas y privadas y en los sistemas analógicos y digitales, terrestre y satelitales. Las miles de horas producidas y emitidas por las cadenas nacionales suponen una dificultad creciente para productores, periodistas, realizadores, editores y documentalistas, a la hora de acceder y procesar una cantidad de información útil que puede ser reciclada en reportajes y documentos históricos en las televisiones. Sin la posibilidad de acceder de manera rápida y confiable a información de interés, sobre el contenido audiovisual almacenado o en transmisión, dichos contenidos se hacen muy difíciles de manejar, es ahí donde aparecen los sistemas de catalogación de medias.

Estos son sistemas informáticos encargado de los procesos de almacenamiento, captura, digitalización y monitoreo de contenidos audiovisuales facilitando el intercambio entre redes de información audiovisual existentes.

El entorno y las posibilidades de uso de estos sistemas es muy variado y cada vez la forma en que se manejan los grandes volúmenes de información en los medios de comunicación tiende a ser soportados en información audiovisual. Para ello existen diversas soluciones de software a nivel mundial que son capaces de procesar y analizar grandes volúmenes de datos con el fin de obtener y cruzar información relacionada con las diferentes temáticas y acontecimientos que mueven el mundo.

En Cuba el centro GEySED perteneciente a la facultad 6, de la Universidad de las Ciencias Informáticas cuenta entre sus proyectos con Video Web. Este presenta una plataforma que entre los servicios más comunes que brinda se encuentra la difusión de videos, ya sea seriales televisivos, películas, videos humorísticos, promocionales o simplemente la transmisión de canales de televisión en vivo y video noticias, entre otros.

Es de gran interés para este proyecto poder saber en secuencias de video noticias en cuales escenas aparecen personas, pues sobre estas se podría realizar luego una catalogación más detallada de las personas que aparecen o sobre su comportamiento. Sin embargo debido a que la cantidad de multimedia a monitorear en esta plataforma es muy grande, se hace difícil e ineficiente catalogar de forma manual las escenas donde aparecen personas. Ante la situación existente se identifica como problema a resolver: **La detección de personas por rostros, en las escenas de video de noticias, en un Sistema de Captura y Catalogación de Medias es muy complejo y poco confiable cuando se realiza por un operador humano.**

Para solucionar dicho problema se tiene como objetivo general de la investigación: **Proponer un algoritmo para la detección de rostros de personas presentes en secuencias de video noticias.** Se plantea como objeto de estudio: **Métodos y algoritmos para la detección de personas por rostros,** delimitando como campo de acción: **Métodos y algoritmos para la detección de personas por rostros en secuencias de video noticias.**

Planteando como idea a defender que: **la implementación del algoritmo para la detección de personas en secuencias de video noticias, facilitará la catalogación de escenas que contienen personas dentro de la plataforma de Video Web.**

Para darle cumplimiento al objetivo general se trazaron las siguientes **tareas de investigación:**

- Caracterizar el estado del arte sobre los métodos y algoritmos de detección de personas por rostros en secuencias de video noticias.
- Seleccionar los métodos y algoritmos para el desarrollo del componente de detección de personas.
- Definición de la tecnología a usar para el desarrollo de los algoritmos.
- Implementar el algoritmo para la detección de personas por rostros en secuencias de video noticias.
- Validar el funcionamiento del algoritmo propuesto.

Con el correcto cumplimiento de las tareas se espera obtener los siguientes **resultados**:

- ❖ Informe de investigación teórico sobre la detección de personas por rostros en secuencias de video noticias.
- ❖ Prototipo funcional que permita detectar escenas donde aparecen personas a partir de la detección de rostros de personas en secuencias de video noticias.

Para obtener los conocimientos necesarios que hagan posible el cumplimiento del objetivo trazado en el trabajo y cumplir las tareas propuestas, se lleva a cabo una investigación en la que se utilizan los siguientes métodos científicos:

Métodos Teóricos:

- ❖ **Analítico-Sintético:** se utiliza en el análisis de los elementos bibliográficos, definiciones y apuntes para la recopilación de información sobre los métodos de detección a partir de bases de conocimientos.
- ❖ **Histórico-Lógico:** ayuda a caracterizar las soluciones existentes estudiando su evolución y facilitando aprovechar puntos en común y conceptos teóricos que sean de relevancia para la realización del algoritmo.
- ❖ **Modelación:** brinda la posibilidad de crear abstracciones para explicar la realidad existente haciéndose visible en el trabajo al crear el modelo de implementación del algoritmo y un prototipo funcional que demuestre el funcionamiento del mismo.

Capítulo1: Fundamentación Teórica.

Introducción

Este capítulo está dirigido a plantear todos los elementos teóricos que sustentan el objeto de estudio y el objetivo de la investigación. Se relacionan todos los conceptos que desde el punto de vista teórico permiten un mejor entendimiento de lo que se plantea en la situación problemática y en el marco del problema en sentido general, con el objetivo de proporcionar mayor cantidad de información al respecto y enfatizar sobre la situación problemática donde coexiste el objeto de estudio.

También se realiza un estudio sobre las soluciones que existen y que brindan una solución parcial o total a la situación problemática. Se realiza un análisis de las mismas con el fin de facilitar la comprensión de la importancia de la investigación, su alcance y su aporte científico.

1.1 Conceptos asociados al dominio del problema.

Imagen digital

La imagen digital es un producto del desarrollo de la informática que tiene como antecesor a la fotografía, (que toma como punto de partida un objeto del mundo real) y a la pintura, (donde la imagen ha sido creada por un artista), en la imagen digital podemos ver incluidos los dos hechos, la originalidad de la imagen cuando es tomada por primera vez, y luego el resultado de compresiones, optimizaciones, filtrados y otros procesos que forman parte del arte digital. (Castro)

Algunos especialistas como Rafael C González y Richard E. Woods, consideran que “una imagen puede ser definida como una función bidimensional $f(m, n)$, donde m y n son coordenadas espaciales, y la amplitud de f en cualquier par de coordenadas (m, n) se llama intensidad o nivel de gris de la imagen en ese punto. Cuando m, n y los valores de amplitud de f son todos finitos y valores discretos, se le llama a la imagen, imagen digital.” (Castro)

De acuerdo con las valoraciones de Rafael C González y Richard E. Woods se llega a la conclusión que una imagen digital es una matriz bidimensional cuyo objetivo es recrear elementos imaginarios como el arte, o reales como el entorno mismo que rodea al hombre. Las imágenes digitales son muy utilizadas hoy en día, hasta el punto de haber extinguido casi en su totalidad a la fotografía

tradicional. Esta es aplicable en casi todos los campos, entre ellos: la fotografía, el arte, publicidad, comercio, medicina, investigación, educación y otras. Su alta aplicabilidad está dada principalmente por ser fácil de procesar, manejar, clonar y transportar. En la figura 1 se muestra un ejemplo de una imagen digital.



Figura 1. Imagen digital.

Píxel: es la unidad más pequeña que encontraremos en las imágenes compuestas por mapa de bits. El píxel es la unidad mínima en que se divide la retícula de la pantalla del monitor y cada uno de ellos tiene diferente color, es decir, es la menor unidad que compone una imagen en una computadora o cualquier medio electrónico, compuesta a su vez por mapas de bits (Laboratory, 1998.). Ver figura 2.



Figura 2. Píxel.

Video

Está constituido por una gran cantidad de marcos de las imágenes que se integran para formar el video. Pueden o no estar acompañadas de sonido y se basan en píxeles.

Video digital

“El video hace referencia a la captación, procesamiento, transmisión y reconstrucción de una secuencia de imágenes y sonidos que representan escenas en movimiento.” (Cecilia A., 2005)

El video digital se puede definir como la representación digital de la señal de vídeo analógico, es una secuencia de imágenes o frames que son almacenadas y reproducidas a una velocidad y en un orden determinado (Figura 3).



Figura 3. Video digital.

1.2 Objeto de Estudio

El objeto de estudio de esta investigación está centrado en el estudio de los algoritmos y los métodos para la detección de personas por rostros.

1.3 Descripción General

Aunque la detección de rostros es un campo de investigación muy activo, todavía se considera una tarea difícil, por causa de la alta variabilidad de la apariencia de los rostros. Los rostros son objetos no rígidos y dinámicos con una diversidad grande en la forma, el color y la textura, debido a múltiples factores como la pose de la cabeza, iluminación (contraste, las sombras), expresiones faciales, las oclusiones (lentes) y otras características faciales (el maquillaje, la barba). La alta variabilidad en la

apariciencia de los rostros, afecta más la detección de rostros que la variabilidad de la identidad del rostro.

Los pocos datos disponibles de las imágenes generalmente no son suficientes para cubrir la variabilidad en apariencia. Además, puede resultar una gran dificultad a la hora de la detección la pose de la cabeza.

En detección de rostros, el propósito es determinar si hay uno o más rostros en una imagen (o video), y si es el caso, devolver su posición y escala. El término localización es empleado cuando hay únicamente un rostro en la imagen. La detección del rostro es un ámbito de la investigación importante en la visión de computadora, porque sirve como un primer paso necesario a cualquier sistema de procesamiento de rostros, como reconocimiento de rostros, seguimiento de rostros, análisis de expresión. La mayor parte de estos métodos asumen, en general, que la región de rostros ha sido localizada perfectamente. Por lo tanto, sus rendimientos dependen de una manera muy significativa en la precisión del paso de detección de rostros. En la Figura4 se puede apreciar la etapa de detección de rostros.

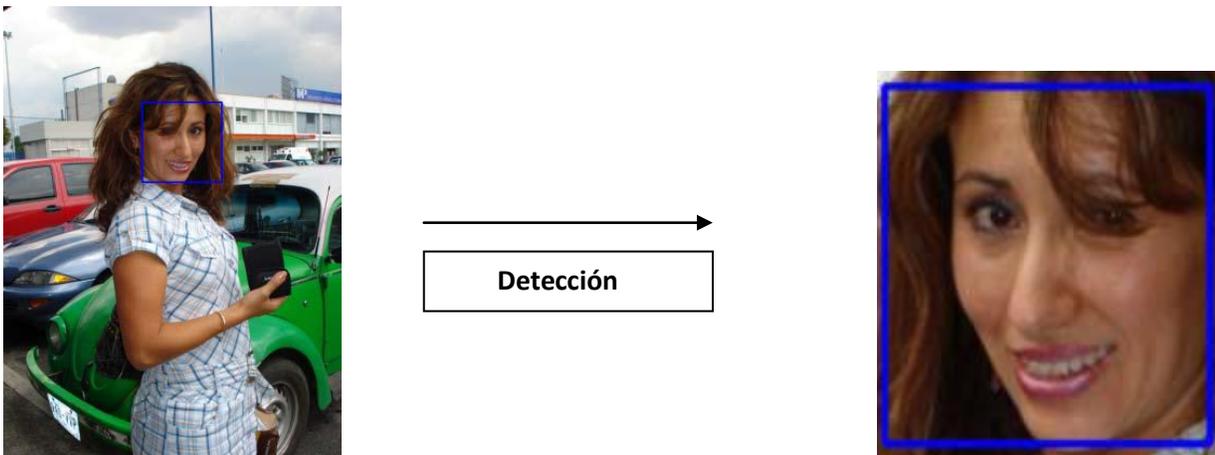


Figura 4. Detección de rostros

1.4 Soluciones Existentes

1.4.1. Detección de rostros

En las aplicaciones donde se realice cualquier tipo de análisis facial, la detección de rostros es una etapa muy importante, de la que depende que luego se pueda realizar un correcto procesamiento de los rostros presentes en las imágenes. Algunas de las actividades que usan la detección de rostros como una etapa base sobre la cual luego se realiza otro tipo de procesamiento se encuentra el reconocimiento de rostros, codificación de video en video conferencias, interfaces hombre-máquina. El objetivo de esta etapa consiste en detectar y localizar la posición de un número indefinido de rostros en una imagen. En general, la detección de rostros es un problema muy complejo ya que los objetos a detectar pueden ser de diferentes colores, expresiones, poses, tamaños relativos o tener condiciones de iluminación diferentes. Se han propuesto numerosos métodos para la detección de rostros, pero básicamente se podrían agrupar en dos grandes categorías: (Rama)

Métodos basados en reglas.

Se establecen relaciones (reglas) entre las diferentes características faciales como por ejemplo la distancia entre ojos o “simetría” del rostro. (Rama)

Métodos estadísticos.

Este tipo de algoritmo no asume ningún tipo de información previa de la tipología de un rostro; sino, que a partir de un conjunto de muestras (imágenes de rostros y de no-rostros) de entrenamiento extraen la información relevante que diferencia un objeto rostro de un objeto no rostro. Este grupo incluye uno de los métodos más referenciados y utilizados actualmente: el detector de rostros AdaBoost. Existen factores que modifican la apariencia de un rostro en una imagen, esto hace la detección de rostros una tarea desafiante. (Rama)

1.4.1.1. Segmentación

Lo primero que se realiza en cualquier proceso de análisis de imagen es la segmentación. Mediante este proceso se divide la imagen en las partes u objetos por la que está formada, como paso previo al análisis de su contenido. La subdivisión es realizada según el nivel que necesite la aplicación en particular, o sea, la segmentación terminará cuando se hayan detectado todos los objetos de interés

para la aplicación. Este proceso de segmentación autónoma, o sea sin necesidad de la intervención de un humano, es una de las tareas más difíciles y complicadas dentro del procesado de imagen. El éxito o fallo del proceso de análisis de una imagen estará dado por la segmentación. Por una parte una segmentación adecuada facilita mucho la solución del problema y una segmentación errónea conducirá a fallos, por lo que se debe poner todo el esfuerzo posible en la etapa de segmentación. Existen dos técnicas de segmentación:

Técnicas Contextuales:

- ❖ Tienen en cuenta las relaciones que existen entre los objetos de la imagen. Tendrá en cuenta tanto las características del píxel como su vecindad con otros. Los algoritmos contextuales están basados en los conceptos de: discontinuidad y similitud.
- ❖ Las técnicas basadas en discontinuidad dividen la imagen detectando cambios abruptos en los niveles de gris de la misma.
- ❖ Las técnicas orientadas a similitud intentan crear regiones uniformes agrupando píxeles que satisfacen ciertos criterios de similitud. El resultado depende fuertemente del criterio y de la definición de conectividad empleada. (Krawczuk)

No Contextuales:

Ignoran las relaciones que existen entre los objetos que aparecen en la imagen. Los píxeles se agrupan de acuerdo a algún atributo común como ser: intensidad, color. (Krawczuk)

Modelado del color de la piel

Existen numerosos trabajos relacionados con el modelado del color de la piel, que abarcan definición implícita, métodos paramétricos y no paramétricos. También se usan diferentes espacios de color, siendo los más usados el RGB (Red, Green, Blue), RGB normalizado y HSV (*Hue, Saturation, Value* – Tonalidad, Saturación, Valor) (E.Woods, 1992). El objetivo de esta tarea es etiquetar cada píxel de la imagen con dos valores posibles: 1 si el píxel posee color de la piel, 0 si el píxel no posee color de la piel. Como resultado se obtiene una imagen binaria donde se distinguen claramente las regiones correspondientes a la piel humana. Este trabajo comenzó usando un modelo explícito del color de la piel en el espacio RGB, que clasificaba un píxel según el siguiente criterio:

Pixel [R, G, B] tiene color de la piel

si: $R > 95$ y $G > 40$ y $B > 20$ y $\max\{R, G, B\} > 15$ y $|R-G| > 15$ y $R > G$ y $R > B$

Sin embargo el método propuesto por Jones y Regh, que consiste en un modelo no paramétrico del color de la piel basado en un clasificador bayesiano, ofrece mejores resultados; por lo cual se determina escogerlo. Como resultado de la aplicación de este método a una imagen se obtiene un mapa de probabilidad de la piel que no es más que una imagen en escala de grises donde cada píxel posee un valor entre 0 y 1 que indica su grado de pertenencia a la clase piel.

Para realizar la segmentación se define un umbral¹, entre cero y uno, y todo píxel cuyo valor sea igual o mayor que dicho umbral es etiquetado como piel y como no piel en caso contrario dando como resultado la imagen binaria descrita anteriormente. El umbral que hemos escogido es 0.7. Un umbral más cercano a 0 daría como resultado muchas zonas etiquetadas con color de la piel sin serlas y uno más alto produciría el efecto contrario. (Castro)

1.4.1.2. Áreas potenciales

El primer paso en un sistema de detección es detectar los píxeles con el color de la piel de la imagen. El resultado de esto es una imagen segmentada por el color de la piel para quitar pequeñas áreas de ruido. Así, las regiones conectadas con el color de la piel son calculadas usando un algoritmo de etiquetado. Una vez que los píxeles que pertenecen a la clase piel son agrupados en una región conexa, aquellos que su dimensión son claramente dimensiones diferentes a la de los rostros son descartadas. Para hacer este descarte son aplicadas 4 diferentes pruebas a las regiones conectadas:

1. Prueba de la mínima y máxima área: Las regiones de piel que son menores que el 1% del total de las áreas de la imagen son descartadas. Las regiones que tienen un área mayor que el 80% del total de la imagen son descartados.

2. Prueba de las regiones largas: Toda el área que bordea un cuadrado, el cual tiene un ancho menor que el 40% del largo, es descartada. Toda el área que bordea un cuadrado, el cual tiene un largo menor que el 40% del ancho, es descartada.

¹ Valor a partir del cual empiezan a ser perceptibles los efectos de un agente físico.

3. Prueba de las regiones esparcidas: Toda región que su área es menor que el 50% del área de su cuadrado es descartada.

4. Prueba de la proporción: Si la proporción largo/ altura de la región es mayor que 1.6, la altura es disminuida hasta que la proporción sea menor que 1.6.

Una vez descartadas las regiones explicadas, las otras regiones son las que se pasan al detector de rostros. (Romero, 2007)

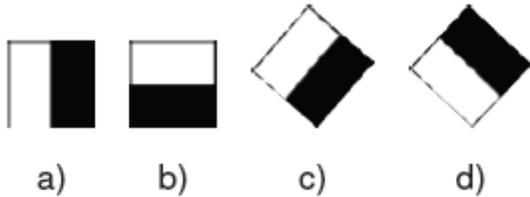
1.4.1.3. AdaBoost

Primero el clasificador es entrenado con imágenes de cientos de muestras de un objeto en particular (ej. un rostro o un carro) llamado muestra positiva, que son redimensionadas al mismo tamaño (ej. 20 X 20), y muestras negativas que son imágenes arbitrarias del mismo tamaño.

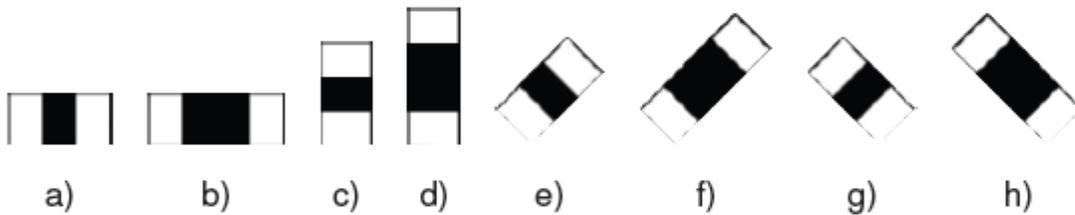
Después que el clasificador es entrenado puede ser aplicado a una región de interés (del mismo tamaño de las muestras usadas en el entrenamiento) de una imagen determinada. El clasificador devuelve 1 si la región es parecida al objeto para el cual fue entrenado y 0 de otra forma. Para buscar el objeto en la imagen completa se mueve una ventana por toda la imagen y chequea en cada lugar usando el clasificador. El clasificador está diseñado para que pueda ser fácilmente redimensionado para que así pueda encontrar el objeto buscado con diferentes tamaños, lo cual es más eficiente que redimensionar la imagen en sí. Para encontrar un objeto de un tamaño desconocido en una imagen el procedimiento debe de hacerse varias veces con diferentes tamaños de ventana.

La palabra “cascade” en el nombre del clasificador significa que la clasificación resultante consiste en varios clasificadores simples (states), que son aplicados sucesivamente a una región de interés hasta que en algún estado el candidato es rechazado o pasa todos los estados. La palabra “boosted” significa que los clasificadores en cada estado de la cascada son en sí mismos complejos y son contruidos de clasificadores básicos usando una de las cuatro diferentes técnicas de boosting (Discrete AdaBoost, Real AdaBoost, Gentle AdaBoost y Logitboost). Los clasificadores básicos son clasificadores de árboles de decisión con al menos 2 salidas. Las características Haar-like son las entradas de los clasificadores básicos, y son calculados como se describe a continuación. El algoritmo actual usa las siguientes características de Haar-like.

1. Características de vértices



2. Características de líneas



3. Características de centro y borde



Figura 5. Características de Haar-like

Las características usadas en un clasificador en particular es especificada mediante la forma de la Figura 5, la posición en la región de interés y tamaño (este tamaño no es el mismo que el usado en los estados de detección, aunque esos tamaños son multiplicados). Por ejemplo, en el caso de la primera línea de característica (2c) la respuesta es calculada como la diferencia entre la suma de los píxeles de la imagen en el rectángulo convergiendo en toda la característica (incluyendo las dos líneas blancas y la línea negra en el medio) y la suma de los píxeles de la imagen en la línea negra multiplicado por 3 para compensar la diferencia en el tamaño de las áreas. Las sumas de los valores de los píxeles en la región rectangular son calculadas rápidamente usando imágenes integrales. (Schulz, 2008)

Este todavía no puede resolver el problema de rotación de rostros en imágenes de rostros reales. Aunque este problema puede ser resuelto mediante la rotación de la imagen y aplicando el método

muchas veces, eso significa un incremento de la complejidad computacional y falsas alarmas. Entonces se necesita desarrollar una técnica especial que mejore la eficiencia del detector.

1.4.1.4. Detección utilizando elipses

Este algoritmo realiza procesamientos de la siguiente forma:

❖ Procesamiento(1):

Localización de áreas del color de la piel utilizando reglas de decisión en el espacio RGB normalizado. Una vez localizadas dichas áreas, se somete la imagen a operaciones morfológicas básicas para remover elementos que por su tamaño no pueden ser rostros y también para mantener las posibles áreas del rostro concatenadas.

❖ Procesamiento(2):

Después de segmentar la imagen empleando la información de color (localización de áreas del color de la piel) se la transforma a escala de grises y en esa nueva imagen se aplica la transformada de Hough para la detección de elipses. Debido a que esta técnica es computacionalmente costosa se la aplica bajo ciertas restricciones, además se emplea la imagen en escala de grises reescalada a la cuarta parte. La transformada de Hough trabaja solo con píxeles de borde por lo que antes de utilizarla se debe realizar la detección de bordes. Para esta tarea se seleccionó el filtro de Sobel.

❖ Procesamiento(3):

Para cada elipse resultante se realiza lo siguiente:

1. Los parámetros de la elipse (coordenadas del centro y longitud de los semi-ejes) son reescalados con referencia al tamaño original de la imagen en escala de grises.
2. Se extrae de esta imagen el área que abarca la elipse y se le trata como una nueva imagen.
3. Para la nueva imagen se detecta el área de los ojos, el área de la boca, se realiza la validación geométrica del triángulo ojos-boca.

4. Si la puntuación obtenida por el triángulo es mayor a un valor umbral se normaliza la imagen en cuanto a orientación del rostro y luego se extrae el área del rostro como una nueva imagen.
5. Se ajusta la intensidad de la imagen del rostro.
6. Se proyecta esta imagen en el espacio de rostros (PCA) y se calcula la distancia entre la proyección y la imagen para confirmar si es un rostro.
7. Si la distancia es menor a un valor umbral el proceso de localización del rostro finaliza, caso contrario se repiten los pasos 1-7 con la siguiente elipse hasta encontrar un rostro o utilizar todas las elipses en cuyo caso se concluye que la imagen no contiene un rostro humano.

❖ Procesamiento(4):

Luego de tener las elipses candidatas, se les aplica el método de Detección Geométrica como se explica en sección anterior (1.4.1.4). (I Jacobson G. Booch, 2000)

1.4.1.5. Redes Neuronales Artificiales para la detección de rostros

Este algoritmo trabaja aplicando una o más Redes Neuronales Artificiales (RNA) directamente a la porción de la imagen entrada y arbitrariamente a sus resultados. Cada neurona es entrenada para una salida de presencia o ausencia de un rostro.

Muchos investigadores tienen la idea que los rostros pueden ser caracterizados directamente en términos de intensidad de píxeles. Estas imágenes pueden ser caracterizadas mediante métodos probabilísticos de un conjunto de imágenes de rostros o de forma implícita mediante una red neuronal u otro mecanismo.

Los parámetros para estos modelos son ajustados de forma automática con imágenes de muestras o manualmente. Algunos autores han tomado estos métodos de extracción de características y aplicando cualquier regla generada manual o automáticamente para evaluar estas características.

Entrenar una red neuronal para la detección es una tarea desafiante dada la dificultad de caracterizar las imágenes que no son rostros. A diferencia del reconocimiento en las cuales las clases a ser discriminadas son diferentes rostros, en la detección las dos clases son imágenes que contienen rostros e imágenes que no los contienen. Es fácil obtener una muestra representativa de imágenes

que contengan rostros, pero muy difícil obtener una muestra representativa de esas imágenes que no contengan rostros. Se evita este problema usando una gran colección de entrenamiento de imágenes que no contienen rostros. El método bootstrap reduce el tamaño de la colección de entrenamiento si es necesario. El uso arbitrario entre múltiples redes y heurísticas para limpiar los resultados significativamente mejora la efectividad del detector

1.5 Elementos computacionales

Para el desarrollo de la investigación se utilizaron algunas herramientas para facilitar la implementación de un prototipo funcional, así como la evaluación de la efectividad del mismo a partir de los resultados arrojados en las pruebas realizadas.

Las ventajas principales de un IDE radican en la automatización de algunas tareas asociadas al desarrollo de aplicaciones y al poseer algunas funcionalidades que facilitan el desarrollo de software disminuyendo así el tiempo de construcción. Por lo que se escogió:

Microsoft Visual Studio.

Es un IDE para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión net 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles. (Microsoft, 27/01/2011)

Visual Studio 2010

Visual Studio 2010 es la versión más reciente de esta herramienta, acompañada por .NET Framework 4.0. Hasta ahora, uno de los mayores logros de la versión 2010 de Visual Studio ha sido el de incluir las herramientas para desarrollo de aplicaciones para Windows 7, tales como herramientas para el desarrollo de las características de Windows 7 (System.Windows.Shell) y la Ribbon Preview para WPF.

Entre sus más destacables características, se encuentran la capacidad para utilizar múltiples monitores, así como la posibilidad de desacoplar las ventanas de su sitio original y acoplarlas en otros sitios de la interfaz de trabajo. Además de esto, aparece una edición que compila las características de todas las ediciones comunes de Visual Studio: Professional, Team Studio, Test, conocida como Visual Studio Ultimate.

OpenCV

OpenCV viene de las siglas Open Source Computer Vision library, es una librería abierta desarrollada por Intel. Esta librería proporciona un alto nivel funciones para el procesamiento de imágenes. Dichas librerías permiten a los programadores crear aplicaciones poderosas en el dominio de la visión digital. OpenCV ofrece muchos tipos de datos de alto-nivel como juegos, árboles, gráficos, matrices, entre otros. Además es open source para poder funcionar en muchas plataformas.

OpenCV permite, operaciones básicas, procesamiento de imágenes y análisis, análisis estructural, análisis de movimiento, reconocimiento del modelo, reconstrucción 3d y calibración de la cámara, Interfaz gráfica y adquisición. Esta implementa una gran variedad de herramientas para la interpretación de la imagen. Es compatible con Intel Image Processing Library (IPL) que implementa algunas operaciones en imágenes digitales. A pesar de primitivas como binarization, filtrado, estadísticas de la imagen, pirámides, OpenCV es principalmente una librería que implementa algoritmos para las técnicas de la calibración (Calibración de la Cámara), detección de rasgos, para rastrear (Flujo Óptico), análisis de la forma (Geometría, Contorno que Procesa), análisis del movimiento (Plantillas del Movimiento, Estimadores), reconstrucción 3D (Transformación de vistas), segmentación de objetos y reconocimiento (Histograma). (Medrano, 23/04/08)

Estructura de Open CV.

Estas bibliotecas se dividen en cinco grandes grupos:

CXCORE: donde se encuentran las estructuras y algoritmos básicos que usan las demás funciones.

CV: donde están implementadas las funciones principales de procesamiento de imágenes.

HighGUI: todo lo relacionado a la interfaz gráfica de OpenCV y las funciones que permiten importar imágenes y video.

ML: que cuenta con algoritmos de aprendizaje, clasificadores y demás.

CvAux: con funciones experimentales. (Medrano, 23/04/08)

¿Por qué utilizar OpenCV?

Ha sido seleccionado gracias a que en cuanto a análisis de movimiento y seguimiento de objetos, ofrece una funcionalidad interesante. Incorpora funciones básicas para modelar el fondo para su posterior sustracción, generar imágenes de movimiento MHI (Motion History Images) para determinar donde hubo movimiento y en qué dirección, algoritmos de flujo óptico.

Lenguaje de Programación

C++

Fue creado a mediados de 1980 por Bjarne Stroustrup bajo el nombre en un principio de C. No fue hasta 1983 que se le llamó C++. El objetivo de su creación era el de extender el conocido lenguaje C permitiéndole a los programadores trabajar con objetos. Poco a poco se le fueron añadiendo facilidades como la programación estructurada y la orientada a objetos. Es un lenguaje imperativo, versátil, potente.

Ha eliminado también algunas de sus desventajas aunque mantiene muchas otras como por ejemplo: sigue muy ligado al hardware subyacente, manteniendo una baja potencia para la programación a bajo nivel. Debido a esto, se le han añadido elementos que le posibilita la programación a alto nivel también. Permite programar desde sistemas operativos hasta compiladores, sistemas de Bases de Datos, procesadores de textos.

Con el tiempo ha evolucionado y contribuido a la aparición del Lenguaje Java. Una de las cualidades de este lenguaje de programación es el de que permite redefinir los operadores y crear nuevos tipos que se comporten como tipos fundamentales. Aunque introduce nuevos operadores y palabras claves para manejar clases, algunas de sus extensiones tienen aplicación fuera del contexto de programación orientada a objetos. (Microsoft, 27/01/2011)

¿Por qué utilizar C++?

Se escogió este lenguaje ya que al utilizar C++, se puede realizar una aplicación distributable sin necesidad de pagar licencia. Además, es un lenguaje multi-nivel, es decir, con él puedes programar

de forma directa tanto hardware (según el sistema operativo que se utilice) como para crear operaciones de tipo Windows los cuales presentan una misma interfaz. Es versátil y mantiene las ventajas del lenguaje C en cuanto al número de operadores y expresiones, flexibilidad, eficiencia y concisión. Brinda una gran facilidad de programación. Con su ayuda pueden programarse desde programas de gran sencillez hasta otros más complicados como sistemas operativos. Presenta la facilidad de que es portable, es decir, un código creado en este lenguaje se puede compilar en cualquier sistema operativo sin necesidad de cambiarle prácticamente el código fuente.

1.6 Conclusiones

Luego de realizar un análisis profundo de las cuestiones teóricas importantes y de las soluciones existentes, se evidenció la necesidad de proponer un algoritmo que cumpla las expectativas del proyecto VideoWeb. Además mediante el estudio de los métodos y elementos computacionales a utilizar, se han sentado las bases para el desarrollo del prototipo funcional en el que se les realizan las pruebas al algoritmo propuesto.

Capítulo 2: Presentación de la Solución Propuesta

Introducción

En este capítulo se presenta la solución a la problemática planteada en un inicio, a partir de un estudio detallado del estado del arte de los métodos y algoritmos eficientes para realizar la detección de personas por rostros, que existen en el mundo. Por lo que se propone un algoritmo compuesto por la combinación de varios métodos, capaz de detectar las escenas en las que aparecen caras de personas en secuencias de video noticias.

1.1. Conceptos Fundamentales

Para un mejor entendimiento de la solución propuesta se proporciona un marco conceptual con las definiciones identificadas, las cuales son:

Persona: Objeto localizado en el video.

Detección: Acción y efecto de detectar.

Características: Rasgos que identifican al objeto.

Seguimiento: Registro de movimientos del objeto a través del tiempo.

Video: Archivo que almacena los flujos de video de una cámara comprimidos en un formato de uso común.

1.2. Detección Geométrica

Detección de Ojos

Para poder encontrar la ubicación de los ojos y la boca dentro de un posible candidato a cara se realizaron mapas derivados directamente de la luma² y de las cromas³ de la imagen original. Un “mapa” es una imagen en tonos de grises que resalta determinadas

² Presencia de brillo en una imagen.

³ Componente de la señal de vídeo que lleva la información del color. Por extensión, el grado o nivel de saturación del color.

características relevantes de la imagen original, para facilitar la detección de algo concreto, en este caso de ojos y boca. Para la construcción de dichos mapas se trabajó en el espacio Y Cb Cr (Luma, Cromo azul, Cromo roja). Para construir el mapa de ojos (EyeMap) se tomaron en cuenta dos puntos:

En los ojos hay poco rojo (bajo Cr) y bastante azul (alto Cb).

En los ojos existen píxeles bien claros así como bien oscuros.

Considerando la primera sentencia, se forma el siguiente mapa de cromas:

$$\text{EyeMapC} = \frac{1}{3} \left(C_B^2 + (255 - C_R)^2 + \frac{C_B}{C_R} \right)$$

Cada término de la suma está normalizado al rango (0; 255). Para implementar la segunda sentencia, se utilizó morfología en escalas de grises. Para esto se utilizó un elemento de estructura hemisférico G. La idea es resaltar aquellas zonas que tienen alto contenido de Y que a su vez tienen vecinos cercanos con bajo contenido de Y. Para esto se dilató la componente Y (imagen de luma) obteniendo así una imagen que resalta las zonas donde hay alto contenido de luma- brillos. Por otro lado se erosionó la Y, resaltando (hacia el 0) las zonas donde hay poco componente de luma. Si el elemento de estructura es lo suficientemente grande, las zonas que tienen alto y bajo componente de luma van a aparecer en las dos imágenes resaltadas (en el primer caso resaltadas positivamente y en el segundo negativamente). Luego se realizó la división entre la imagen dilatada y la erosionada obteniendo así el mapa de ojos considerando la luma. En resumen las operaciones realizadas son las siguientes:

$$\text{EyeMapC} = \frac{Y(i,j) \oplus G_s(i,j)}{Y(i,j) \ominus G_\sigma(i,j) + 1}$$

El tamaño del elemento de estructura utilizado es una medida del mínimo tamaño de ojo a encontrar en la imagen, se estima como:

$$\sigma = \frac{\sqrt{W * H}}{2F_E}$$

Siendo:

- ❖ W: Ancho de la imagen (Máscara)
- ❖ H: Alto de la imagen (Máscara)
- ❖ FE: Cociente entre máximo tamaño de ojo sobre el tamaño promedio de cara. Se toma
FE = 7 píxeles.
- ❖ \oplus : Dilatación.
- ❖ \ominus : Erosión.

Para juntar ambos mapas, primero se normaliza al rango (0; 255) y luego se combina mediante la multiplicación (AND) y el resultado fue enmascarado y nuevamente normalizado al rango (0; 255). Para decidir cuáles son las regiones de ojos, se realizó un umbralizado adaptivo. El umbral de decisión es una combinación lineal entre el promedio del mapa y el máximo valor del mapa, es decir:

$$\text{Umb} = \alpha * \text{promedio (EyeMap)} + (1 - \alpha) * \text{Max (EyeMap)}$$

Siendo un parámetro α a ajustar. Finalmente se obtiene un mapa binario con todos los píxeles que están por encima del umbral. Para terminar esta etapa se realiza un filtrado morfológico (mediante un elemento de estructura circular) para eliminar aquellas regiones que no tienen un mínimo de tamaño, aplicándole primero apertura y luego cerradura, obteniendo así el cierre de la apertura del mapa de ojos binario. Luego se realizó una etapa de etiquetado, donde se identificaron aquellas regiones candidatas a ser ojos.

Una vez que los dos ojos son detectados, la posición de las cejas puede ser localizada dentro de una pequeña región arriba del centro del ojo. Seguidamente puede ser aplicado un mapa para buscar la ceja a lo largo de una columna del iris a la mitad de cada ceja. Algunas de las características de la imagen frontal son invariables, las cuales no cambian con la expresión facial, mientras que hay otras características variables que son las que varían cuando cambia la expresión del rostro.

DetECCIÓN DE LA BOCA

Para construir el mapa de boca (MouthMap) se tomó en cuenta un único punto:

En la boca hay más componente roja comparada con la componente azul, que en el resto de la cara. Observando lo anterior y el hecho de que en la boca hay baja respuesta a $C_r = C_b$, se supone el siguiente MouthMap basado únicamente en la información de las cromas:

$$\text{MouthMap} = C_R^2 \left(C_R^2 - \eta \frac{C_R}{C_B} \right)$$

Siendo η el cociente entre la media de C_R^2

$$\eta = 0.95 \frac{\text{promedio}(C_R^2)}{\text{promedio}\left(\frac{C_R}{C_B}\right)}$$

Todas las componentes de las ecuaciones anteriores son normalizadas a (0; 255).

Finalmente se realizó una etapa de enmascarado (con la máscara de cara) y luego se dilató (utilizando un elemento de estructura hemisférico) buscando regularizar el mapa y resaltar las zonas más claras.

Umbralizado y Etiquetado

Para decidir cuáles son las regiones de boca, se realizó un umbralizado adaptivo similar al realizado en el mapa de ojos. El umbral de decisión es una combinación lineal entre el promedio del mapa y el máximo valor del mapa, donde los coeficientes de ponderación son determinados de forma experimental. Finalmente obtenemos un mapa binario con todos los píxeles que están por encima del umbral. Para terminar esta etapa se realiza un filtrado morfológico para eliminar aquellas regiones que no tienen un mínimo de tamaño, aplicándole primero apertura y luego cerradura, obteniendo así el cierre de la apertura del mapa de ojos binario.

A continuación se realizó un etiquetado de las regiones encontradas como candidatos a boca, para luego facilitar el cálculo de algunas de sus características (área, centroides, ubicación).

Restricciones Geométricas: Validación

Por último se realizó una etapa que combina los resultados encontrados en las etapas anteriores con el fin de validar si la región de piel considerada es una cara o no. En el caso en que se considere como cara, se debe indicar la mejor ubicación de los ojos y la boca. Para poder realizar lo anterior se forman todas las combinaciones posibles entre los candidatos a ojos y boca hallados (si tenemos n bocas, m ojos entonces hay n combinaciones de m tomadas de a dos ternas posibles). A cada combinación se denomina candidato. Para cada candidato, que supere un conjunto mínimo de restricciones se calcula un valor, que contempla que tan buen candidato es.

Con cada par de ojos y cada boca se forma un triángulo cuyos vértices son los centroides de las regiones asociadas a los ojos y boca en consideración.

Primero se aplica una etapa, que es llamada de mínimas condiciones, en la cual si un candidato no las cumple es eliminado. Ésta consiste en:

- ❖ El triángulo formado por los tres centroides(2 ojos, 1 boca) debe ser agudo (no puede tener ninguno de sus ángulos mayor a 90)
- ❖ El triángulo debe tener un área mínima de 100 píxeles (esta restricción está asociada al tamaño mínimo de cara a detectar)

Luego para cada candidato que cumpla las restricciones anteriores se le calculan los siguientes Valores:

- ❖ Valor de simetría y orientación.
- ❖ Valor de simetría de tamaño de ojos.
- ❖ Valor de coherencia de tamaño de cara.

Los valores están formados básicamente por términos de la forma de campanas tipo gaussianas. Dichas campanas están centradas en el valor considerado ideal y con un ancho del óvulo tal que contemple que tan significativo es separarse de dicho valor ideal.

Valor de simetría y orientación

Para el cálculo de este valor se tuvo en cuenta dos aspectos:

Simetría: que tan isósceles es el triángulo formado (Ángulo q 1)

Orientación: que tan inclinado respecto a la vertical se encuentra. (Ángulo q 2)

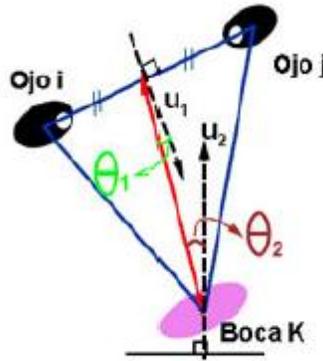


Figura 6. Triángulo de verificación

Para el cómputo del valor se realizó lo siguiente:

$$ScOrienta = \prod_{k=1,2} e^{-3 \sin^2 \frac{\theta_k}{2}}$$

Está formado básicamente por Gaussianas que dependen del cuadrado del seno del ángulo. La idea es favorecer a las caras simétricas y a las que se encuentran muy cerca de la vertical (mirando hacia arriba).

Valor de simetría de tamaño de ojos

Se implementó un valor que contempla positivamente el hecho de que las regiones formadas por cada ojo tengan áreas similares. El valor es una campana de la forma:

$$ScAreaOjos = e^{-\frac{Area_i - Area_j}{\min(Area_i, Area_j)}}$$

Valor de coherencia de tamaño de cara

Mediante un breve estudio estadístico se llegó a la conclusión de que el área del triángulo formado por los ojos y la boca, es el 10% del área total de la cara. Si se aproxima el área de la cara a la de la máscara, se puede implementar un valor para favorecer a los candidatos que cumplan esto, es decir:

$$ScTamano = e^{-\sqrt{Area(Triangulo)-Area(Mask)}}$$

Finalmente los tres valores son agrupados en un único valor mediante una combinación lineal de la siguiente manera:

$$Sc = 0.5 * ScOrienta + 0.25 * ScAreaOjos + 0.25 * ScTamano$$

Se tuvo en cuenta que el valor más determinante para la validación de una cara es el de orientación y simetría. Por último se devuelve como válido aquel candidato que tenga mayor valor y que su valor sea mayor a 0.6 Si no existen candidatos con valor mayor a 0.6 se considera que el candidato no es cara.

1.3. Propuesta de métodos para la detección de rostros

Proponer un método para la detección de rostros se hace hasta cierto punto difícil dado los múltiples caminos que se pueden tomar mediante la combinación de los métodos estudiados como muestra la Figura 7, se mencionaran algunas de estas combinaciones para ayudar a confirmar la propuesta.

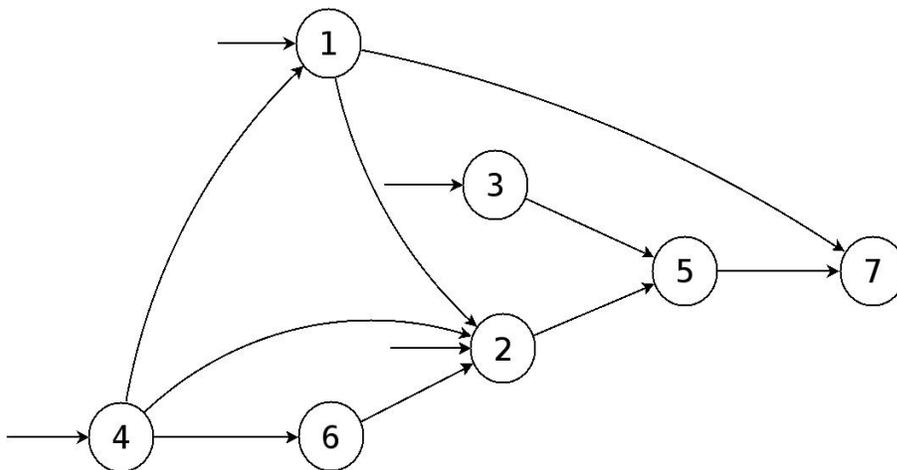


Figura 7. Combinaciones de métodos.

A continuación se muestra la leyenda del grafo en la tabla 1 y los caminos del mismo que pueden ser tomados para la selección en la tabla 2.

Tabla 1. Leyenda

1	AdaBoost
2	AdaBoost para detectar ojos y boca
3	Modelo del color para detectar ojos y boca
4	Segmentación de la imagen por el color de la piel
5	Validación del triángulo formado por los ojos y la boca
6	Detección de rostros candidatos mediante elipses
7	Detección de rostros

Tabla 2. Caminos

A	1 - 7
B	1 - 2 - 5 - 7
C	4 - 1 - 7
D	4 - 1 - 2 - 5 - 7
E	3 - 5 - 7
F	2 - 5 - 7
G	4 - 6 - 2 - 5 - 7
H	4 - 2 - 5 - 7

Seleccionar la combinación **E** (Modelo del color para detectar ojos y boca → Validación del triángulo formado por los ojos y la boca → Detección de rostros) implica que no se puede utilizar la detección de las áreas potenciales mediante la segmentación de la imagen por el color de la piel, ya que la imagen segmentada descartaría el color blanco de los ojos y el rojo fuerte de los labios, colores que necesita este modelo matemático para detectar los ojos y boca.

Esto podría significar la detección incorrecta de rostros en áreas que pudieron haber sido eliminadas para mejorar la detección. Además, tanto este como las combinaciones **F** (AdaBoost para detectar ojos y boca → Validación del triángulo formado por los ojos y la boca → Detección de rostros), **G** (Segmentación de la imagen por el color de la piel →

Detección de rostros candidatos mediante elipses → AdaBoost para detectar ojos y boca → Validación del triángulo formado por los ojos y la boca) y **H**(Segmentación de la imagen por el color de la piel → AdaBoost para detectar ojos y boca → Validación del triángulo formado por los ojos y la boca → Detección de rostros) realizan la validación entre todos los pares de ojos detectados con todas las bocas, que implicaría un modelo matemático muy complicado.

Los métodos **B** (AdaBoost → AdaBoost para detectar ojos y boca → Validación del triángulo formado por los ojos y la boca → Detección de rostros) y **H**(Segmentación de la imagen por el color de la piel → AdaBoost para detectar ojos y boca → Validación del triángulo formado por los ojos y la boca → Detección de rostros) resuelven el problema de validar todos los pares de ojos con todas las bocas ya que enmarcan estas en regiones similares pero esto trae consigo un procesamiento adicional buscando primero un candidato a cara y luego comprobarlo mediante el método geométrico.

Para la selección del método a proponer se tomaron en cuenta algunos elementos como son la velocidad y efectividad de la detección así como el conocimiento del equipo de desarrollo de las herramientas necesarias para la implementación del método propuesto. De los métodos estudiados el más veloz es el AdaBoost y por su excelente efectividad, se seleccionó como propuesta.

Existen varias combinaciones con este método como se explicó anteriormente, de estos la combinación de métodos **A** (AdaBoost → Detección de rostros) y **C** (Segmentación de la imagen por el color de la piel → AdaBoost → Detección de rostros) cumplen con los elementos necesarios para ser seleccionados.

1.4. Conclusiones

Con el desarrollo de este capítulo se ha logrado un mayor entendimiento de las características y restricciones que debe poseer el algoritmo propuesto para cumplir con los requisitos del cliente. Luego del análisis de los métodos estudiados se ha realizado una propuesta del mismo, donde se precisaron los elementos fundamentales y los aspectos fundamentales por los que fueron escogidos.

Capítulo 3: Pruebas al algoritmo propuesto.

Introducción

En el presente capítulo se realizan pruebas de rendimiento al algoritmo, donde queda demostrado la efectividad del mismo en las escenas dónde aparecen rostros de personas.

Se les realizaron pruebas a una muestra de 20 videos de noticias ver Figura8.

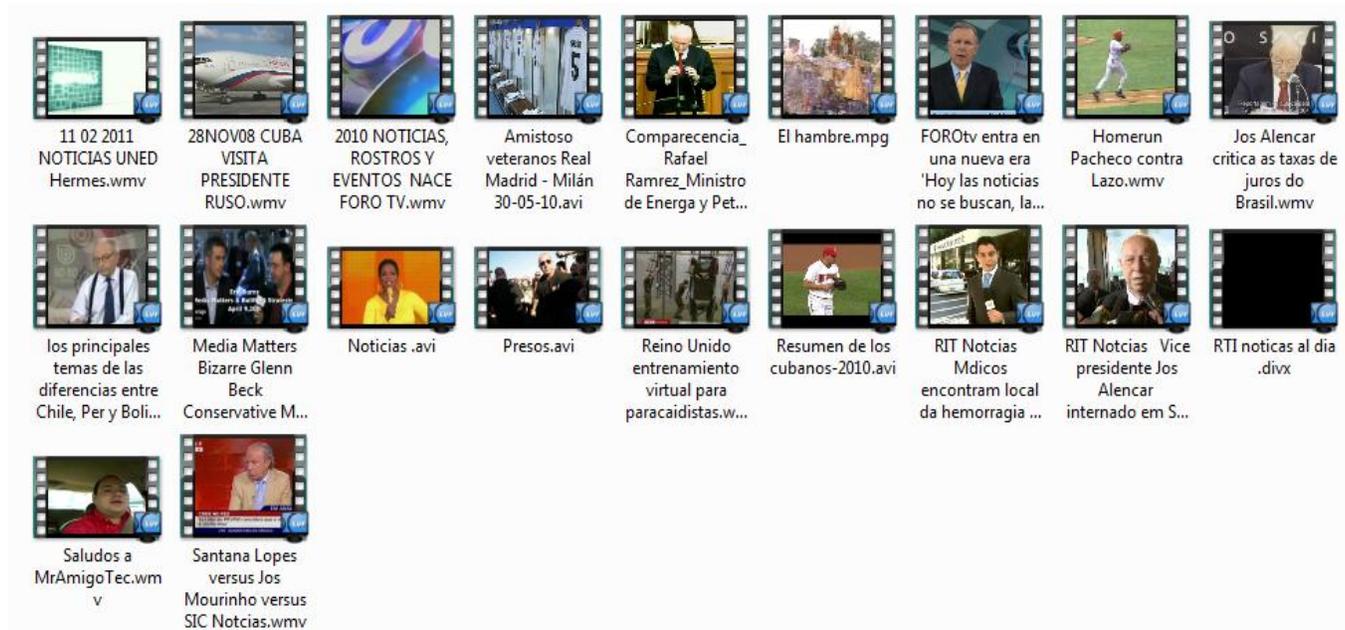


Figura 8 Videos a los que se les realizaron las pruebas

Para probar así la validez de la solución propuesta, a continuación se presentan los resultados obtenidos donde:

Falso Positivo (FP): Son las áreas detectadas como rostros que no lo son.

Falso Negativo (FN): Son las áreas que son rostros y no fueron detectadas.

CR: Es la cantidad de rostros.

El por ciento de efectividad es calculado de la siguiente manera:

$$P = 20 - \frac{(FP + FN) * 100}{CR}$$

Tabla 3 Resultados de Efectividad

FP	FN	CR
182	19	440



Figura 9 Falsos Positivos.



Figura 10 Falsos Positivos.

Esta prueba arrojó un 54,3% de efectividad, este porcentaje es influenciado mayormente por la cantidad de FP. Como se evidencia en la Figura 9 esta efectividad podrá ser mejorada considerablemente mediante la búsqueda de áreas potenciales para lo cual se necesita un proceso de segmentación por el color de la piel que sea factible.

Esta prueba arrojó un 80% de efectividad resultado satisfactorio pues los videos fueron tomados con el fin de realizar la detección lo que hace que esta sea más precisa. Luego de estas pruebas se evidencia que para los videos de noticias sin realizar la segmentación la efectividad no es buena, por lo que se segmentaron los mismos en busca de áreas potenciales, y se realizó otra prueba que eliminó varios falsos positivos., Con este resultado se realiza una tercera prueba descartando los rostros que ocupan menos del 5% de la imagen, que arrojó un 80% de efectividad aumentando considerablemente. A continuación se muestran los resultados.

Tabla 4 Totales

FP	FN	CR	Porcentaje
35	59	463	80



Figura 11 Efectividad del algoritmo luego de aplicada la segmentación



Figura 12 Efectividad del algoritmo luego de aplicada la segmentación.

3.1 Prototipo Funcional

Para poder comprobar la efectividad del algoritmo propuesto fue necesario implementar un prototipo funcional, a continuación se presentan algunas imágenes de este.

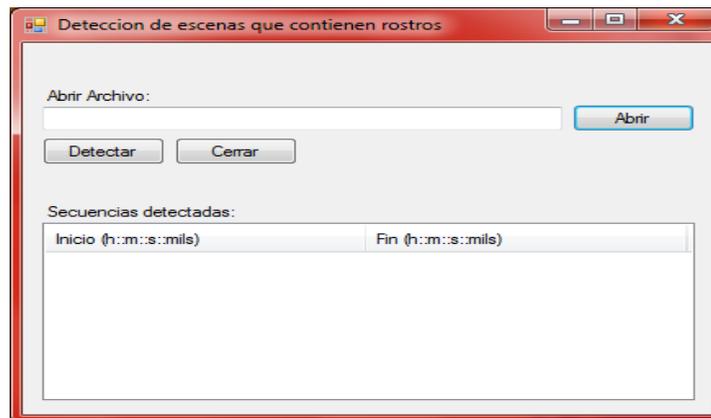


Figura 13 Detección de escenas que contienen rostros.

Al presionar la opción Abrir se tendrá la posibilidad de abrir cualquier video de noticias al cual se le podrá realizar la detección de rostro.



Figura 14 Buscando ubicación del video.

Una vez seleccionado a cual se le realizará la detección, se aplica la opción detectar y es ejecutada la acción.

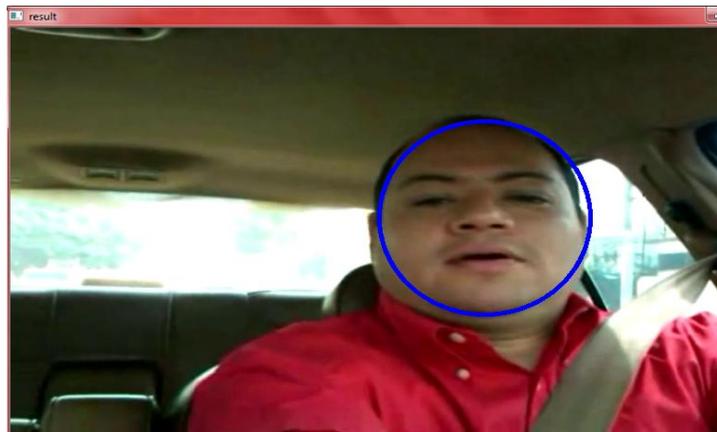


Figura 15 Detección de rostros.

Una vez detectados los rostros presentes en las escenas del video, en la parte inferior se guardan las secuencias en las que fueron detectadas, el tiempo inicial en que comienza y termina la detección, las

horas, minutos, segundo y milisegundos. Terminada la detección se tiene como resultado las secuencias de video en las que aparecen rostros de personas.

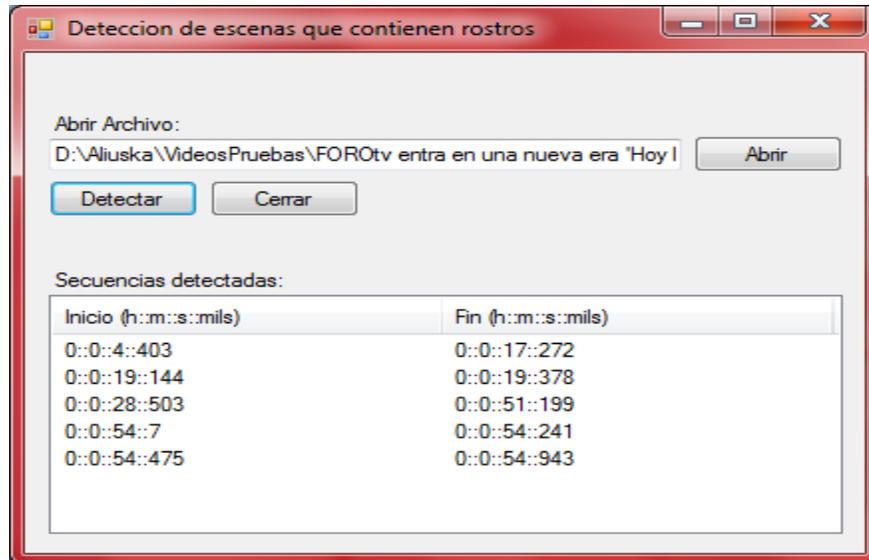


Figura 16 Secuencias detectadas.

3.2 Conclusiones

En este capítulo se implementó un prototipo funcional al cual se le hicieron pruebas de integración mostrando un correcto funcionamiento. En las pruebas con videos se evidenció la necesidad de reducir las áreas de búsqueda mediante la segmentación, pues esto daría una solución mucho más segura para la detección de rostros de personas.

Conclusiones Generales

Durante el desarrollo de la presente investigación, mediante el cumplimiento de las tareas y objetivos propuestos se llegó a las siguientes conclusiones:

- ❖ Existen una variedad de algoritmos para la detección de rostros, siendo la mayoría de ellos muy complejos computacionalmente y para su implementación.
- ❖ Para elaborar el algoritmo propuesto se usó una combinación de métodos que resuelven el problema de la detección de rostros, ajustándolos a su uso en video.
- ❖ Para el desarrollo y prueba del algoritmo se usaron librerías existentes muy conocidas por su eficacia y por ser libres.
- ❖ El algoritmo propuesto permite la detección de las escenas en las que aparecen rostros de personas, siendo más efectivo cuando estos aparecen de manera frontal.

Recomendaciones

La propuesta realizada no se considera un producto acabado, si no que se le pueden agregar nuevas funcionalidades que lo complementen para una futura identificación, para ello se recomienda:

- ❖ Hacer entrenamientos con muestras de rostros rotados lateralmente para lograr una detección de perfil.
- ❖ La búsqueda de un segmentado en cuanto al color de la piel más eficiente que el actual para lograr un mejor resultado en la extracción de las áreas potenciales.

Bibliografía

2007. Aprender Photoshop. *Aprender Photoshop*. [En línea] 2007. [Citado el: 10 de diciembre de 2011.] http://aprender-photoshop.blogspot.com/2007_05_01_archive.html.

Bello, Richard Nolberto Rojas. 2009. *Identificación de características relevantes*. Universidad Autónoma de Madrid : s.n., 2009.

Bustamante, Paul , y otros. 2007. *Aprenda C++ Básico como si estuviera en primero*. San Sebastian : s.n., 2007.

Càceres, Francisco González. 2002. *Reconocimiento de rostros humanos basados en una red neuronal de retropropagación*. Santiago de Chile : s.n., 2002.

Castro, José Luis Alba. *Detección de caras y localización de características*. s.l. : Dpto. de Teoría de Señal e Comunicaciones. 36310 Vigo..

Cecilia A., Germán C., Mauricio D., y Matías M. 2005. Detección de ojos. [En línea] 2005. <http://iie.fing.edu.uy/investigacion/grupos/gti/timag/trabajos/2005/caras/>.

CERECEDA, CARLOS MARCELO ARAVENA. 2009. *ESTUDIO DE MÉTODOS PARA DETECCIÓN Y SEGUIMIENTO DE ROSTROS FRONTALES EN IMÁGENES DIGITALES: TRANSFORMADA ELÍPTICA LINEAL Y MOLDES PSO*. SANTIAGO DE CHILE : s.n., 2009.

Delac, Marian Stewart Bartlett Kresimir. *Recent advances in face recognition*. Vienna : Tech Education and Publishing KG.

Detección automática del color de la piel en imágenes bidimensionales basado en el análisis de regiones. [En línea] [Citado el: 10 de Diciembre de 2010.] <http://www.escet.urjc.es/~jjpantrigo/PFCs/MemoriaPielFeb05.pdf..>

E.Woods, Richard. 1992. *Digital Image Processing*. 1992.

Escuntar, Karla Mariana Romero. 2006. *Reconocimiento de rostros en tiempo real utilizando una red neuronal*. Quito : s.n., 2006.

Flores, R. De la Rosa. 2007. *Procesamiento de imágenes digitales*. Mexico : s.n., 2007.

Hoiem, Derek. march 31, 2004. *Adaboost*. march 31, 2004.

I Jacobson G. Booch, and J. Rumbaugh. 2000. *El Proceso Unificado de Desarrollo de Software*. 2000.

JosuéFortis. 2005. *Diseño Web y Multimedia 1*. 2005.

- Krawczuk, Walter M. Martínez.** *Segmentación de imágenes médicas.*
- Kriegman., Joao P. Hespanha Peter N. Belhumeur and David J. 2002.** *Eigenfaces vs. fisherfaces: Recognition.* 2002.
- Laboratory, Cambridge Research. 1998..** *Statistical color models with application to skin detection.* 1998.
- Lienhart, Alexander Hartmann Rainer.** *Classifying images on the web automatically.* Santa Clara, California : Intel Labs, Intel Corporation 2200 Mission College Boulevard.
- Mateos, Gines García. 2007.** *Procesamiento de Caras Humanas Mediante Integrales Proyectivas.* Murcia : s.n., 2007.
- Maydt, Rainer Lienhart and Jochen.** *An extended set of haar-like features for rapid object detection.* Santa Clara : Intel Labs, Intel Corporation. CA 95052.
- Medrano, Raúl Igual Carlos. 23/04/08.** *Tutorial de OpenCV.* 23/04/08.
- Microsoft. 27/01/2011.** *Cursos de Desarrollo Microsoft.* 27/01/2011.
- Rainer, Luhong Liang. 2006.** *Final report.* 2006.
- Rama, Francesc Tarrés Antonio.** *Un nuevo método para la detección de caras basado en integrales.* Barcelona, España : Dept. Teoria del Senyal i Comunicacions Dept. Teoria del Senyal i Comunicacions - Universitat-Politécnica de Catalunya, Barcelona,.
- Romero, Karla. 2007.** *Reconocimiento de rostros utilizando una red neuronal.* Quito : s.n., 2007.
- Schulz, Daniel. 2008.** *Detección de ojos mediante adaboost.* 2008.
- Segmentación de imágenes de color.* **Bàez Rojas, j y Guerrerob, M. DICIEMBRE 2004.** 50, Mexico : s.n., DICIEMBRE 2004, Vol. 6.
- Sochman, Jiri Matas and Jan. 2006.** *AdaBoos.* Prague : Centre for Machine Perception Czech Technical University, 2006.
- Urueña, William Ardila y Guevara, Marta Lucía. Junio de 2008..** *Detección de rostros en imágenes.* Pereira. : Scientia et Technica Año XIV,, Junio de 2008. 38.
- Viola, Michael J. Jones Paul.** *Robust real-time face detection.* s.l. : Mitsubishi Electric Research Laboratory.
- Vitores, Raúl Sánchez. 2004.** *Sistemas de Reconocimiento Facial .* 2004.

Referencias Bibliográficas

Castro, José Luis Alba. *Detección de caras y localización de características.* s.l. : Dpto. de Teoría de Señal e Comunicaciones. 36310 Vigo..

Cecilia A., Germán C., Mauricio D., y Matías M. 2005. Detección de ojos. [En línea] 2005.
[http://iie.fing.edu.uy/investigacion/grupos/gti/timag/trabajos/2005/caras/.](http://iie.fing.edu.uy/investigacion/grupos/gti/timag/trabajos/2005/caras/)

E.Woods, Richard. 1992. *Digital Image Processing.* 1992.

I Jacobson G. Booch, and J. Rumbaugh. 2000. *El Proceso Unificado de Desarrollo de Software.* 2000.

Krawczuk, Walter M. Martínez. *Segmentación de imágenes médicas.*

Laboratory, Cambridge Research. 1998.. *Statistical color models with application to skin detection.* 1998.

Medrano, Raúl Igual Carlos. 23/04/08. *Tutorial de OpenCV.* 23/04/08.

Microsoft. 27/01/2011. *Cursos de Desarrollo Microsoft.* 27/01/2011.

Rama, Francesc Tarrés Antonio. *Un nuevo método para la detección de caras basado en integrales.* Barcelona, España : Dept. Teoria del Senyal i Comunicacions Dept. Teoria del Senyal i Comunicacions - Universitat Politècnica de Catalunya, Barcelona,.

Romero, Karla. 2007. *Reconocimiento de rostros utilizando una red neuronal.* Quito : s.n., 2007.

Schulz, Daniel. 2008. *Detección de ojos mediante adaboost.* 2008.

Glosario de Términos

- Librería** Conjunto de subprogramas que contienen códigos y datos que brindan servicios a otros programas y ayuda a los programadores a desarrollar el software. No necesita ser modificado. El código que contienen las librerías, se añaden al programa principal cuando se genera.
- IDE** Integrate Development Enviroment:
Entorno de desarrollo integrado. Herramienta que se usa para facilitar el desarrollo de software.
- RGB** Red, Green and Blue:
Rojo, verde y Azul. Es un modelo de color en el cual es posible representar un color mediante la mezcla de estos tres colores primarios.
- HSV** Hue, Saturation, Value:
Modelo de color basado en tres parámetros tono, saturación y valor.
- FP** Falso Positivo.
- FN** Falso Negativo.
- Frames** Un fotograma o cuadro, una imagen particular dentro de una sucesión de imágenes que componen una animación.