



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA DE SOFTWARE PARA
RECONOCIMIENTO FACIAL EN VIDEO

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL ELECTRICISTA

DARÍO EDUARDO VILLALÓN DE LA VEGA

PROFESOR GUÍA:
CLAUDIO PÉREZ FLORES

MIEMBROS DE LA COMISIÓN:
PABLO ESTÉVEZ VALENCIA
NELSON MORALES OSORIO

SANTIAGO DE CHILE
SEPTIEMBRE 2012

RESUMEN DE MEMORIA PARA
OPTAR AL TITULO DE INGENIERO
CIVIL ELECTRICISTA.
POR: DARÍO VILLALÓN DE LA
VEGA
FECHA : 24/09/2012
PROFESOR GUÍA: CLAUDIO PÉREZ

“DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA DE SOFTWARE PARA RECONOCIMIENTO FACIAL EN VIDEO”

Este trabajo aborda el reconocimiento facial en video, cuya problemática se ha convertido en los últimos años en un tema altamente recurrente en diversos ámbitos tales como sistemas de seguridad, plataformas de redes sociales, aplicaciones de marketing, etc.

A modo de contextualización se presenta una revisión del estado del arte en el ámbito de las metodologías de reconocimiento facial y sus principales áreas de aplicación hoy en día. También se abordan los avances actuales en reconocimiento de rostro en video, detallando iniciativas privadas y de comunidades abiertas.

El objetivo general planteado corresponde al diseño, implementación y validación de una plataforma de software para el reconocimiento facial en un archivo de video, utilizando métodos ya establecidos y validados en el campo de la biometría, como lo son el algoritmo Haar para detección de rostros y el algoritmo PCA Eigenface para el reconocimiento facial.

La problemática de fondo tiene que ver con la utilización de estos algoritmos en un escenario como el de un video, en el cual no necesariamente existe una actitud colaborativa de parte de las personas, aportando gestos, poses y luminosidad muy cambiantes, lo cual dificulta la comparación entre los rostros detectados y el rostro objetivo que se desea identificar.

El sistema desarrollado se compone principalmente de 4 módulos funcionales: módulo de ingreso de imágenes del rostro objetivo para el entrenamiento del sistema; módulo de ingreso del video a procesar; módulo del proceso de reconocimiento y módulo de entrega de resultados.

Adicionalmente se desarrolló un módulo con una herramienta que facilita la generación de la referencia de verdad o ‘*Ground Truth*’ para el rostro objetivo en el video de interés, con el fin de evaluar de la efectividad del sistema.

Para la ejecución de pruebas, se seleccionaron tres rostros objetivo a partir de los cuales se construyó una base de imágenes con 100 fotografías por cada rostro las que fueron recolectadas desde Internet. También se seleccionó desde Internet un video de alta resolución para cada rostro.

Los resultados de las pruebas son presentados mediante el análisis de curvas ROC y cálculo del índice AUC para cada curva.

Las conclusiones obtenidas apuntan a una alta dependencia de la calidad de la información de entrenamiento del sistema para lograr una aceptable capacidad de reconocimiento, que en este caso alcanzo a un 73% en el caso más favorable.

Las líneas de trabajo futuras propuestas apuntan a la utilización de algoritmos alternativos para reconocimiento facial, y a la inclusión de funciones adicionales de pre-proceso de imágenes, con el fin de mejorar la estandarización de los rostros a procesar.

Con amor, a:

Marcela

Mis hijos Florencia y Vicente

Mi Madre y mi Padre

Agradecimientos

Deseo agradecer en primer lugar a mi profesor guía, Sr. Claudio Pérez por la oportunidad y paciencia entregada para el desarrollo de mi trabajo de tesis.

Deseo también agradecer a los profesores Pablo Estévez y Nelson Morales por su amable colaboración como miembros de mi comisión de titulación.

Agradezco también al profesor Jorge López por los valiosos consejos entregados en el Taller EL69E.

INDICE

Capítulo 1	10
1. Introducción	10
1.1. Contexto General	10
1.2. Objetivos	10
1.3. Hipótesis de trabajo	11
1.4. Organización del Documento	11
Capítulo 2	13
2. Contextualización	13
2.1. El Reconocimiento Facial	13
2.2. Reconocimiento Facial en video	14
2.3. Aplicaciones y Desarrollo Actual	15
2.4. Algoritmo PCA Basado en Eigenface [6][16]	17
Capítulo 3	23
3. Implementación	23
3.1. Descripción General de la Solución	23
3.2. Descripción detallada del sistema	26
3.3. Contrastación de Resultados con Referencia ‘Ground Truth’	34
Capítulo 4	37
4. Pruebas y Resultados	37
4.1. Datos de prueba	37
4.2. Pruebas	45
4.3. Evaluación de las Pruebas	46
4.4. Resultados	46
4.5. Discusión de Resultados	55
Capítulo 5	58
5. Conclusiones y Líneas de trabajo Futuras	58

Referencias	60
ANEXO A: Especificaciones Ambiente de Desarrollo y Pruebas.	63
ANEXO B: Funciones OpenCV utilizadas para el Procesamiento de Imágenes	65
ANEXO C: Algoritmo de Detección Facial	67
ANEXO D: Manual de Usuario del Sistema.....	73
ANEXO E: Formatos de Archivos de Salida.....	88

INDICE DE TABLAS

Tabla 1: Proyectos de Reconocimiento Facial	17
Tabla 2: Interpretación de VP, VN, FP y FN.	35
Tabla 3: Pruebas Rostro A.....	45
Tabla 4: Pruebas Rostro B.....	45
Tabla 5: Pruebas Rostro C.....	46
Tabla 6: Criterio de intervalos para valores de AUC.....	46
Tabla 7: Posibilidades de clasificación de los cuadros de video de respecto el rostro objetivo.	85

INDICE DE FIGURAS

Figura 1: Transformación de una imagen NxN en un vector 2Nx1.....	18
Figura 2: Diagrama Funcional del Sistema: etapas de entrenamiento y reconocimiento en video	25
Figura 3: Interfaz de Módulo de Entrenamiento.....	26
Figura 4: Pre-procesamiento de Imagen de Entrenamiento.....	27
Figura 5: Interfaz de Módulo de video.	28
Figura 6: Módulo de Reconocimiento.	29
Figura 7: Pre-procesamiento de un cuadro de video para reconocimiento de rostro.....	30
Figura 8: Interfaz de módulo de resultados.	31
Figura 9: Interfaz de módulo de configuración.	33
Figura 10.: Interfaz herramienta Ground Truth.	34
Figura 11: Diagrama secuencial del procesamiento de información de una prueba.....	35
Figura 12: Ejes de movimiento posibles de un rostro frontal.	38
Figura 13: Muestra de video de prueba A (Obama).	39
Figura 14: Muestra imágenes del rostro A on-wild.	40
Figura 15: Muestra imágenes del rostro A extraídas desde el video.	40
Figura 16: Muestra de video de prueba B (Putin).....	41
Figura 17: Muestra imágenes del rostro B on-wild.	42
Figura 18: Muestra de imágenes del rostro B obtenidas del video.....	42
Figura 19: Muestra de video de prueba C (Sarkozy).....	43
Figura 20: Muestra de imágenes del rostro C on-wild.	44
Figura 21: Muestra de imágenes del rostro C obtenidas del video.....	44
Figura 22: Curvas ROC pruebas P1.1, P1.2, P1.3, P1.4.....	47
Figura 23: Curvas ROC pruebas P1.5, P1.6, P1.7, P1.8.....	48
Figura 24: Curvas ROC pruebas P2.1, P2.2, P2.3, P2.4.....	49
Figura 25: Curvas ROC pruebas P2.5, P2.6, P2.6, P2.7.....	50

Figura 26: Curvas ROC pruebas P3.1, P3.2, P3.3, P3.4.....	51
Figura 27: Curvas ROC pruebas P3.5, P3.6, P3.7, P3.8.....	52
Figura 28: Comportamiento AUC set de pruebas A.....	53
Figura 29: Comportamiento AUC set de prueba B.	54
Figura 30: Comportamiento AUC set de prueba B.	55
Figura 31: Características Haar	68
Figura 32: Etapas metodología Haar.	69
Figura 33: Imagen Integral.	69
Figura 34: Filtros Haar rotados, trasladados y con cambios de Escala.....	70
Figura 35: Convolución del filtro Haar con una imagen integral.....	71
Figura 36: Clasificador en cascada.....	72
Figura 37: Módulo de inicio	73
Figura 38: Módulo de entrenamiento.	74
Figura 39: Módulo de video.	76
Figura 40: Módulo de Configuración	78
Figura 41: Módulo de Procesamiento.....	81
Figura 42: Módulo de Resultados.....	82
Figura 43: Módulo Herramienta <i>Ground Truth</i>	86
Figura 44: Módulo Ayuda	87
Figura 45: Contenido archivo Ground Truth.....	90
Figura 46: Contenido Archivo Consolidado.txt.....	91

Capítulo 1

1. Introducción

1.1. Contexto General

Desde los inicios del siglo XXI el desarrollo tecnológico asociado a la captura y procesamiento de imágenes digitales ha experimentado un notable avance. Dispositivos cada vez de menor costo y con una mayor calidad se han convertido en un fenómeno masivo a tal punto que hoy en día elementos tales como teléfonos celulares ponen a nuestro alcance la capacidad de adquisición y almacenamiento de imágenes y videos de gran calidad.

De similar manera esta tecnología ha transformado campos tales como la medicina, la seguridad, el marketing y otros, incorporando el procesamiento de imágenes como una valiosa fuente de información. Con ello, el video y fotografía digital en particular asociado al rostro humano se han transformado en un formato de almacenamiento masivo de datos, frente al cual surge la necesidad de contar con sistemas de detección y reconocimiento de rostros automatizados, fiables y con alta performance.

En este contexto se aborda el desarrollo del presente trabajo de título, el cual pretende plantear un aporte en el campo del procesamiento de imágenes, mediante el desarrollo de una plataforma de software cuya función principal es la de lograr el reconocimiento de rostros dentro un archivo de video digital.

1.2. Objetivos

1.2.1. Objetivo General

Mediante el planteamiento de una metodología de diseño y el uso de ciertos elementos de apoyo, se plantea como objetivo general la implementación y validación de una plataforma de software que sea capaz de realizar la identificación de un rostro objetivo en un archivo de video digital.

1.2.2. Objetivos Específicos

- a) Lograr un diseño que represente una propuesta de solución para el objetivo general planteado.
- b) Determinación de la plataforma y herramientas de apoyo adecuadas para el desarrollo del sistema.
- c) Programación y operación de la plataforma de reconocimiento facial en base a las especificaciones definidas.
- d) Definición y ejecución de pruebas para determinar la efectividad del sistema.
- e) Proponer caminos para desarrollo futuro.

1.3. Hipótesis de trabajo

Mediante la utilización de los algoritmos Haar Cascade y PCA Eigenface para detección y reconocimiento de rostro respectivamente, es posible programar una plataforma software para reconocimiento de facial en video digital.

1.4. Organización del Documento

La memoria está estructurada de la siguiente forma:

En el capítulo 2 de *Contextualización* se detallan aspectos relacionados con el estado del arte en el campo específico de la identificación de rostros en video digital, además de conceptos y conocimiento esencial para el entendimiento de la memoria.

En el capítulo 3 de *Implementación* se describe el sistema desarrollado, incorporando el diseño previo y los detalles de la implementación propiamente tal.

En el capítulo 4 de *Pruebas y resultados* se describen los set de datos utilizados y las pruebas ejecutadas para determinar la validez del sistema. También se presentan los resultados obtenidos y una discusión en torno a ellos.

En el capítulo 5 de *Conclusiones y Líneas de trabajo futuras* se verifica el cumplimiento de los objetivos iniciales planteados, así como también se entrega una visión, a partir de los resultados de las pruebas, del campo de uso de la plataforma desarrollada y su fiabilidad. Se plantean también líneas de trabajo futuras focalizadas como puntos de mejora para el sistema desarrollado.

En los *Anexos 1, 2, 3, 4 y 5* se presenta información específica respecto los elementos de apoyo utilizados, tales como: algoritmos para detección y reconocimiento de rostro; librerías y funciones para procesamiento de imágenes; especificaciones de los ambientes de desarrollo y pruebas; Manual de usuario del sistema; Información de los archivos de datos generados por el sistema.

Capítulo 2

2. Contextualización

En este capítulo se presenta una revisión general del estado del arte en el campo del reconocimiento facial, respecto los principales métodos actualmente desarrollados y su uso tanto en el ámbito académico como industrial.

También se presenta una revisión más específica en el ámbito del reconocimiento facial en video, abordando los principales problemas que enfrentan, los beneficios y las principales aplicaciones.

Por último, se presenta desarrollo teórico respecto al método PCA basado en Eigenfaces el cual es utilizado en la implementación de esta tesis.

2.1. El Reconocimiento Facial

En la vida cotidiana normalmente no nos sorprendemos de nuestra capacidad para reconocer rostros, incluso de personas a las cuales no vemos desde hace años. Distintas edades, razas, lejos o cerca, con mala o buena iluminación, sin importar la circunstancia, siempre tenemos la capacidad de reconocer un rostro en base a nuestra experiencia previa o recuerdos.

Sin embargo, esta problemática no es tan simple cuando se intenta realizar por medios computacionales, utilizando hardware y software que tienen limitaciones y que dependen de la calidad de información previamente ingresada respecto del rostro que se desea identificar y de la metodología utilizada.

Dado lo anterior, el reconocimiento facial forma parte del área de estudio denominada Biometría, cuyo propósito consiste en el desarrollo de métodos automatizados para la identificación o verificación de personas mediante características físicas o de comportamiento.

En particular, las primeras iniciativas de reconocimiento facial automatizado se remontan a los inicios de las década del 60 [20], con la compañía norteamericana *Panoramic Research* en Palo

Alto, California, la cual, con financiamiento del Departamento de Defensa de EEUU, desarrolló un sistema capaz de extraer características de un rostro mediante la asignación de coordenadas a distintos elementos tales como boca, nariz, ojos, cejas, etc., las cuales eran comparadas con una base de datos que contenía las coordenadas previamente ingresadas de distintas personas. La computadora calculaba la distancia entre los registros de la base de datos y los datos de la fotografía a identificar.

El siguiente gran hito en este campo se produce a partir del año 2001 con el ataque de las torres gemelas en EEUU, donde la biometría cobra un gran auge [21]. Los sistemas de seguridad utilizados en aeropuertos, y lugares públicos en general comienzan a requerir sistemas más sofisticados y confiables. También se produce un notable avance en el desarrollo tecnológico del hardware requerido, avanzando hacia dispositivos con cada vez más alta resolución y de costos cada vez más reducidos.

Hoy en día, los campos de uso del reconocimiento facial han desbordado el ámbito de la seguridad, llegando hacia áreas tales como: el marketing (personalización de ofertas en grandes tiendas y supermercados); redes sociales (*Google, Facebook* incorporan la búsqueda de personas en base al reconocimiento facial); aplicaciones para gestión de información multimedia (*Picasa* de *Google, Apple* Iphoto incorporan esta tecnología); etc. [22] [23]

El reconocimiento facial en video corresponde a un campo de la biometría relativamente nuevo, donde los principales estudios y aplicaciones se han desarrollados en los últimos siete u ocho años [24]. Los sistemas de video-vigilancia han abarcado el mayor uso de esta tecnología, la cual presenta algunos desafíos que hasta hoy en día no han sido del todo resueltos [25], derivados principalmente del carácter no colaborativo de las personas en el contexto de un video, representado principalmente por la diversidad de iluminación, gesto y pose de los rostros, factores que no contribuyen al buen desempeño de los diversos algoritmos y métodos de reconocimiento facial [26].

2.2. Reconocimiento Facial en video

Este campo se ha convertido en los últimos años en unos de las áreas de la biometría con gran desarrollo. Grandes empresas a nivel mundial, tales como *Google, Facebook e IBM* están realizando importantes inversiones con el fin de contar con este tipo de tecnología [22][23]. Sin embargo los resultados con que actualmente se cuenta no son del todo alentadores puesto que la

exactitud que proveen estos sistemas aún sigue siendo deficiente [27]. Entre las causas que explican lo anterior se encuentran [27]:

- Calidad de video: Un video puede estar grabado en uno o varios ambientes, con una iluminación inadecuada o cambiante.
- Personas no son colaborativas: no necesariamente las personas ayudan o están conscientes que están siendo grabadas en un video para ser reconocidas.
- Imágenes de rostro de baja escala: debido a condiciones de adquisición de la imagen, el tamaño de la cara puede ser reducido, lo cual afecta a la exactitud del proceso.

Hay aplicaciones en las que un sistema de reconocimiento facial en video puede obtener buenos resultados, tales como en sistemas de control de acceso y cajeros de bancos, donde el video se adquiere en un ambiente relativamente controlado, con un tamaño de la cara adecuado y lo más importante, el aspecto colaborativo de las persona. En estos casos, el reconocimiento en video ofrece algunas ventajas respecto el basado en imágenes estáticas:

- El video proporciona más información, ya que se cuenta con más cuadros para elegir en cual realizar la clasificación.
- El video proporciona continuidad temporal, permitiendo eventualmente reutilizar la información de clasificación obtenida de los cuadros de alta calidad para procesar los cuadros de baja calidad.
- El video permite hacer un seguimiento de las caras, así los cambios de pose y las expresiones faciales se pueden compensar.

2.3. Aplicaciones y Desarrollo Actual

En la actualidad existen diversos productos, tanto comerciales como de acceso gratuito, que solucionan o proveen facilidades en el área de la detección/reconocimiento de rostros en fotografías o video. Luego de una investigación realizada en Internet, se logró la identificación de 15 proyectos que configuran el estado del arte en el ámbito del reconocimiento facial:

Nombre	Tipo de Producto	Descripción
Picassa	Gratuito	Programa que permite el inventariado de todos los

		archivos gráficos de un computador, su clasificación y orden, e incluye además herramientas de edición y retoque fotográfico. Posee una funcionalidad de indexación a partir del reconocimiento de rostro. Durante el años 2013 incorporará un sistema de indexación y búsqueda de videos utilizando el reconocimiento facial.
iPhoto	Comercial	Provisto por Apple con la compra de un computador Macintosh, es un programa que permite importar, organizar, editar, imprimir y compartir fotos digitales. Posee una funcionalidad de indexación a partir del reconocimiento de rostro.
Facebook	Gratuito	Sitio Web para redes sociales que provee la facilidad de reconocimiento de rostro (Face Recognition) para agrupar caras similares juntas y automáticamente sugerir amigos que se deberían 'taggear' con estos. Durante el 2012 ha incorporado el reconocimiento facial en videos.
Diva3D	Comercial	Provisto por Alpha Tec Ltd., es un programa comercial que provee diversas herramientas para procesamiento de imágenes y video, entre la cuales se halla la de reconocimiento de rostro en video. Su principal campo de uso es el de Seguridad.
PittPatt	Comercial	Provisto por Pittsburgh Pattern Recognition, es un Kit comercial de desarrollo para lenguaje C/C++. Provee bibliotecas para desarrollar aplicaciones con Reconocimiento Facial. (http://www.pittpatt.com). Esta empresa fue recientemente adquirida por Google para potenciar su plataforma de redes sociales Google+.
FaceEnforce	Comercial	Provisto por la empresa SYBULA, es un robusto sistema para reconocimiento de rostro basado en los últimos avances de captura 3D de imágenes faciales combinados con el método AURA para reconocimiento de patrones.
VeriLook SDK	Comercial	Provisto por la empresa NeuroTechnology, corresponde a una suite de soluciones que entre muchas funcionalidades realiza reconocimiento de rostro bajo distintos escenarios. Se provee de soluciones de software ya desarrolladas así como de herramientas para realizar desarrollo e integración a otros sistemas.
FastAccess SDK	Comercial	Provisto por la empresa SensibleVision, es un paquete de software orientado al desarrollo de soluciones de control de acceso y autenticación basadas en el reconocimiento de rostro.
Malic	Proyecto de código	Es un proyecto de código abierto dedicado al desarrollo de librerías para C/C++ en al ámbito del

	abierto	reconocimiento de rostro, el cual esta aun en una etapa muy básica.
Gaborboosting	Comercial	Programa de orientación científica que aplicable al desarrollo de metodologías para reconocimiento de rostro, basado en wavelets Gabor and algoritmo AdaBoost.
Luxand FaceSDK	Comercial	Provisto por la empresa Luxand, es un set de librerías para el desarrollo bajo Microsoft Visual C++, C #, VB, y Delphi, que permite crear aplicaciones Web, Windows, Linux y Macintosh con funcionalidades de reconocimiento facial e identificación biométrica basada en la cara.
FaceVACS-SDK	Comercial	Provisto por la empresa Cognitec, Set de herramientas orientado a integrar tecnología de reconocimiento facial en aplicaciones existentes. Proporciona una API fácil de integrar y distribuir a las aplicaciones cliente.
Faint	Proyecto de código abierto	Provisto por la Universidad de Oldenburg, es un proyecto de código abierto que desarrolla un cuadrowork Java para la detección de rostros y tecnologías de reconocimiento facial. Se basa en diferentes tipos de filtros y plug-in. Posee una interfaz gráfica además de un navegador de fotos integrado que permite a los usuarios aplicar la detección de rostros y el proceso de reconocimiento de imágenes personales.
Libface	Proyecto de código abierto	Proyecto de código abierto cuyo objetivo es desarrollar un set de algoritmos para reconocimiento facial de enfoque Eigenface.
Ares	Proyecto de código abierto	Proyecto de código abierto cuyo fin es desarrollar un set de algoritmos para reconocimiento facial basado en el uso de las librerías OpenCV para procesamiento de imágenes.

Tabla 1: Proyectos de Reconocimiento Facial

2.4. Algoritmo PCA Basado en Eigenface [6][16]

PCA Eigenface es un método de reconocimiento de rostros basado en el llamado sub-espacio de imágenes. En este contexto, una imagen de $n \times n$ píxeles es representada como una matriz de dimensiones $N \times N$ píxeles, donde cada píxel posee un valor de intensidad diferente (valores entre 0 y 255) que se obtiene al transformar la imagen del tipo RGB (rojo, verde y azul) a escala de grises. Las imágenes están formadas por coordenadas: $I(x,y)$.

Debido a que las imágenes serán de rostros humanos, éstas poseen similitudes entre sí, y se pueden describir a través de una representación en un sub-espacio cuya dimensión es mucho menor a N .

La idea principal del método es encontrar los vectores que mejor representen la distribución de las imágenes, los cuales definen un sub-espacio correspondiente a las imágenes de rostros de seres humanos. Ese sub-espacio es llamado espacio de rostros.

El proceso comienza con una imagen de rostro $I(x,y)$ que se puede representar en dos dimensiones $N \times N$ (matriz de valores de intensidad de 8-bits).

Una imagen puede ser considerada como un vector de dimensiones N^2 . Una típica imagen de 256 x 256 pixeles puede ser un vector de una dimensión de 65.536 .

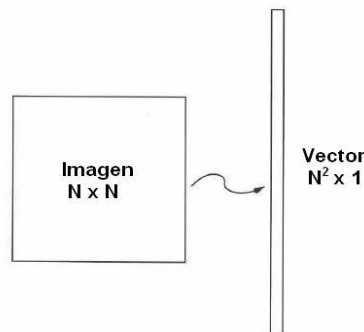


Figura 1: Transformación de una imagen $N \times N$ en un vector $N^2 \times 1$.

El objetivo del análisis de componentes principales es encontrar los vectores que mejor almacenen la distribución de las imágenes de rostros en el espacio completo de imágenes. Estos vectores definen el sub-espacio de imágenes.

El primer conjunto de imágenes de rostros con el que se va a trabajar y a formar el espacio de rostros puede ser $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ el cual corresponde al conjunto de entrenamiento de imágenes de uno o más individuos contra el cual se va a comparar la imagen de entrada o test.

Este es el punto de partida para el Análisis de Componentes Principales (PCA) el cual busca encontrar el grupo de vectores que mejor describan la distribución de dichos datos. El promedio de este conjunto se define por:

$$\Psi = \frac{1}{M} \sum_{j=0}^M \Gamma_j . \quad (2.1)$$

Luego se calcula la diferencia ϕ_i , entre el vector que representa cada rostro con el rostro promedio:

$$\phi_i = \Gamma_i - \Psi . \quad (2.2)$$

Con estos vectores diferencia se forma la matriz A de tal manera que:

$$A = [\phi_1, \phi_2, \dots, \phi_M] .$$

Entonces se calcula la matriz de covarianza C, de dimensiones $N^2 \times N^2$ mediante la siguiente fórmula:

$$C = \left(\sum_{i=0}^{M-1} \phi_i \phi_i^T \right) / (N - 1) = AA^T . \quad (2.3)$$

En este punto se debería encontrar los vectores y valores propios de la matriz de covarianza (llamados Eigenvectores y Eigenvalores respectivamente), pero esto es una tarea demasiado compleja debido al inmenso tamaño de esta matriz. Es por esto que se adopta una forma simplificada para calcular estos vectores y valores propios. Esta forma simplificada se fundamenta en que dado que el número de imágenes de entrenamiento es usualmente mucho menor que el número de píxeles de la imagen, tan solo existirán M en lugar de N^2 vectores propios significativos, los demás son asociados a valores propios igual a cero.

Es por eso que se calculan los vectores y valores propios de la matriz L que tiene un tamaño $M \times M$ y es igual a:

$$L = A^T A . \quad (2.4)$$

Una vez que se obtiene el vector V, que contiene los M vectores propios (EigenVectores) $v_j (j=1,2,\dots,M)$ de la matriz L, se pueden obtener los vectores propios de la matriz de

covarianza mediante una combinación lineal de los vectores diferencia ϕ_i y los vectores propios de la matriz L. Para esto se utiliza la siguiente ecuación:

$$U = [u_1, u_2, \dots, u_M] = [\phi_1, \phi_2, \dots, \phi_M] [v_1, v_2, \dots, v_M] = AV \quad (2.5)$$

En la práctica no es necesario utilizar los M vectores principales para construir las *eigenfaces* sino un conjunto un poco menor, $M' < M$, lo que permite obtener una mayor compresión. Para esto se debe seleccionar los M' vectores propios más significativos, por lo cual se ordena los vectores propios de mayor a menor y se utiliza tan solo los M' primeros vectores propios que son los que forman el espacio de rostros. Estos vectores pueden ser redimensionados al tamaño original de la imagen N x N y es lo que se conoce como *eigenfaces*.

Los eigenfaces son empleados para clasificar la imagen de entrada, y cada nueva cara o imagen de entrada es transformada en su componente eigenface, para luego ser comparada con los eigenfaces del conjunto de entrenamiento. Se busca encontrar las direcciones de $\{U_i\}$ donde las desviaciones se encuentran concentradas, es decir donde los datos transformados estén des-correlacionados. Los valores propios proveen una medida de la cantidad de desviación en dirección al vector U_k , donde $\{U_i\}$ es la base ortonormal del sub-espacio de las muestras, es decir del conjunto de entrenamiento, con $i = 0, 1, \dots, M-1$.

Se obtienen unas nuevas imágenes, proyectándolas en el espacio del rostro con la siguiente operación:

$$\omega_k = u_k^T [\Gamma - \Psi] \quad \text{donde } k = 1, \dots, M' \quad (2.6)$$

Aquí se puede seleccionar un $M' < M$ que corresponderá a los eigenvectores que poseen los eigenvalores más grandes, ya que son los que tienen asociados los Eigenfaces que contienen mayor información de la imagen representada. Para el proceso de reconocimiento, una imagen de entrada ingresa al sistema; esta se transforma a escala de grises, luego se convierte en un vector columna, se normaliza y se calcula la matriz de pesos de las imágenes del conjunto de entrenamiento mediante la expresión:

$$\Omega^T = [\omega_1, \dots, \omega_M] . \quad (2.7)$$

Esto describe la contribución de cada eigenface, en representación de la imagen de entrada. El peso correspondiente a la imagen de entrada será:

$$\Omega_{in} = U^T \text{EntradaNormalizada} . \quad (2.8)$$

Luego se calcula la **distancia euclidiana** entre los vectores correspondientes al conjunto de entrenamiento y la imagen de entrada, de la siguiente manera:

$$E_k = \|\Omega_{in} - \Omega_k\| . \quad (2.9)$$

Determinando de esta forma la clase que minimiza la distancia euclidiana y se compara contra determinado UR.

En el análisis PCA se puede reconstruir cualquiera de las imágenes del conjunto de muestras así.

$$\Gamma_j = \Psi + \sum_{i=0}^{M-1} w_{ij} u_i . \quad (2.10)$$

De esta forma, el algoritmo de reconocimiento de rostro utilizando PCA basado en Eigenfaces sigue los siguientes pasos:

- 1.- Almacenar un conjunto de imágenes de entrenamiento de diferentes personas, pudiéndose tener subconjuntos de imágenes para cada persona que contengan diferentes posturas, condiciones de iluminación, etc.
- 2.- Crear una matriz formada por la imagen de entrada (o test) y las ya almacenadas en la base de datos. Mediante el proceso arriba descrito, se calculan los *eigenvectores* mediante la matriz de covarianza.

3.- Una vez obtenidos los vectores característicos se comparan las distancias entre el vector que representa a la imagen original con el resto.

4.- Establecido un umbral de reconocimiento (en adelante, UR) a priori, si el menor valor del paso anterior es menor que el UR, la imagen de la cara de entrada es considerada como conocida, si es mayor, se considera desconocida. De aquí se desprende que los valores de las distancias dependen en cierta medida del tamaño de la base de datos, puesto que la matriz de covarianza y los vectores característicos son calculados a partir de la matriz formada por la imagen de entrada y las ya almacenadas, por lo que el UR ha de ser dinámico y adaptarse según la variación de las distancias entre caras, o lo que es lo mismo, según el tamaño de la base de datos que almacena las imágenes de las caras.

Capítulo 3

3. Implementación

Este capítulo describe la implementación realizada, comenzando con una descripción funcional general del sistema, para posteriormente detallar cada etapa o módulo. También se describen brevemente los algoritmos utilizados para la detección e identificación facial y los elementos de apoyo empleados para la programación.

3.1. Descripción General de la Solución

3.1.1. Plataforma de desarrollo

Para el desarrollo del sistema se utilizó lenguaje C++, considerando las siguientes características básicas requeridas para el objetivo planteado:

- Necesidad de contar con un lenguaje potente para procesamiento matemático y gráfico.
- Orientado a objetos con el fin de dar modularidad a las distintas etapas del procesamiento de imágenes y visualización de resultados.
- Programación de alto y bajo nivel.

Para determinar el IDE (ambiente de desarrollo) más adecuado, se analizaron varias alternativas principalmente del ámbito código abierto, descartando el uso de productos dependientes del pago de licencia, principalmente para no limitar la eventual continuidad del desarrollo futuro sobre el presente trabajo.

Los IDE evaluados como alternativas fueron: CodeBlocks, Dev-C++, WxDev-C++ y Eclipse.

Se eligió WxDev-C++ (ver Anexo1), básicamente debido a que:

- Es uno de los IDE de código abierto que posee más años de existencia
- Cuenta con una evolución de versiones que permite suponer un nivel de madurez
- Contiene nativamente en compilador MinGW (que corresponde al GCC para Windows)
- Cuenta con una biblioteca llamada WxWidgets que provee una completa gama objetos para el manejo de interfaces gráficas.

Para el manejo y procesamiento de imágenes se determinó el uso de la biblioteca de código abierto para desarrollo de visión artificial OpenCV (Ver anexo 1).

Los aspectos más importantes que justifican esta elección son los siguientes:

- Originalmente desarrollada por Intel, el proyecto cuenta con más de 11 años de existencia y constante evolución, por lo cual es un proyecto que posee un alto grado de madurez y constante actualización.
- Provee una gran variedad de herramientas para procesamiento directo de imágenes (ve anexo2).
- Provee una gran variedad de funciones desarrolladas especialmente para el manejo de la metodología de Eigenface.
- Posee librerías de funciones compatibles con el lenguaje de programación C++.
- Amplio soporte y literatura en Internet debido a su uso muy difundido.
- Si bien la última versión liberada corresponde a la V.2.4 de Julio del 2012, para este trabajo se optó por utilizar la versión anterior, V.2.3, que a juicio de la comunidad usuaria posee un grado de madurez mayor cuanto a problemas reportados y cerrados.

3.1.2. Arquitectura general del sistema.

El sistema implementado se basa en un modelo subdividido en módulos funcionales, los que componen un esquema de reconocimiento facial en video. Dichos módulos son los siguientes:

- Información de entrada (imágenes de entrenamiento de rostro objetivo y video de entrada)
- Entrenamiento del sistema en base a imágenes del rostro objetivo
- Configuración del sistema
- Proceso de reconocimiento facial en el video.
- Entrega de resultados. Contrastación con referencia *ground truth*.
- Entrega de datos finales, para generación de curvas ROC e índice AUC.

En la figura 2 se detalla la arquitectura general del sistema.

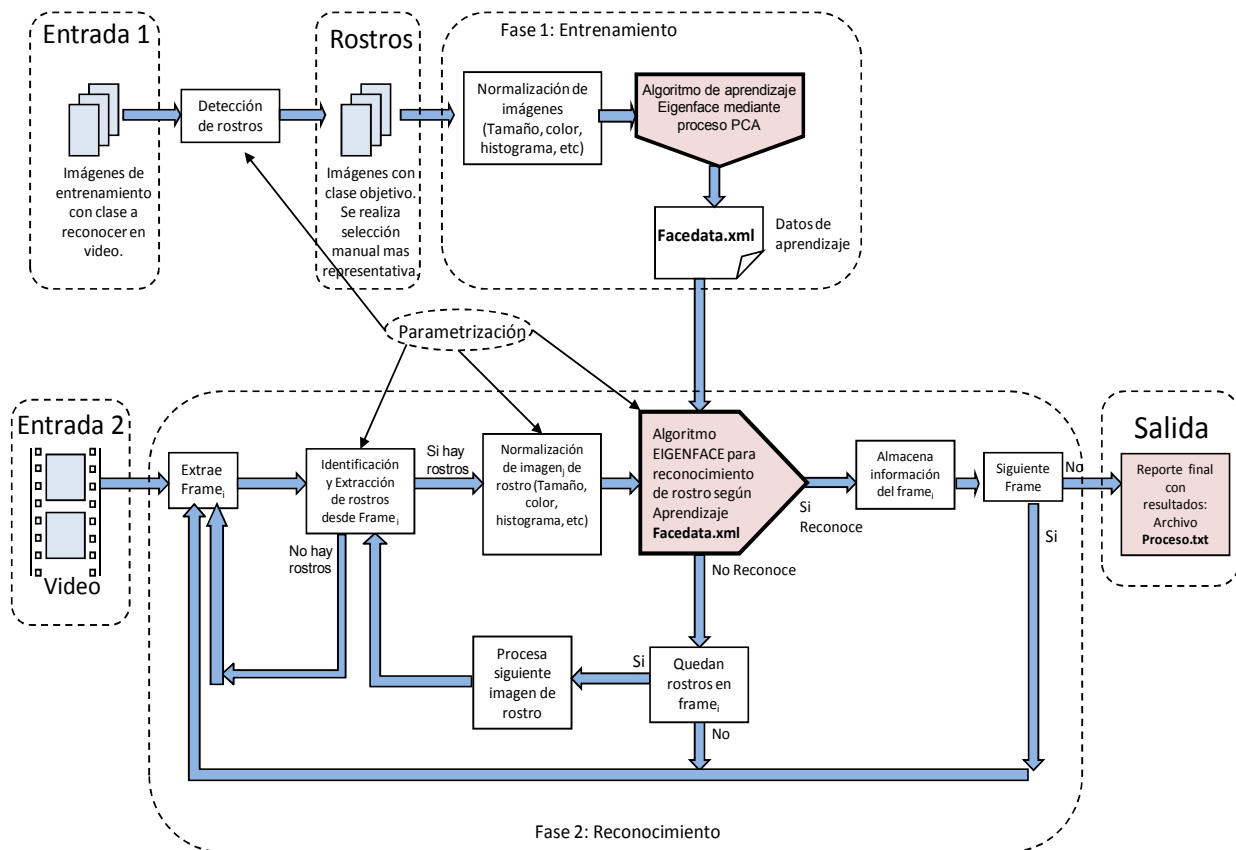


Figura 2: Diagrama Funcional del Sistema: etapas de entrenamiento y reconocimiento.

3.1.3. Método de detección de Rostro

Para programar la detección de rostros tanto en la etapa de entrenamiento del sistema como en el proceso de localización de rostros en los cuadros de video, se utilizó el algoritmo llamado Haar Cascade, el cual se detalla su funcionamiento en el Anexo 3. Para la implementación en lenguaje C++ se utilizó las funciones disponibles en la biblioteca OpenCV especificadas en el Anexo 2

3.1.4. Método de Reconocimiento de Rostro

Para programar el reconocimiento facial, se utilizó el algoritmo llamado PCA (acrónimo de Principal Component Analysis) basado en Eigenface, para el cual se detalla su funcionamiento en 2.4. Para la implementación en lenguaje C++ se utilizó las funciones disponibles en la biblioteca OpenCV especificadas en el Anexo 2.

3.2. Descripción detallada del sistema

3.2.1. Módulo Entrenamiento

Es el encargado de permitir el ingreso de las imágenes con las que se realizará el entrenamiento del sistema. Todas deben pertenecer al rostro que se desea identificar en el video, considerando distintas poses y gestos.

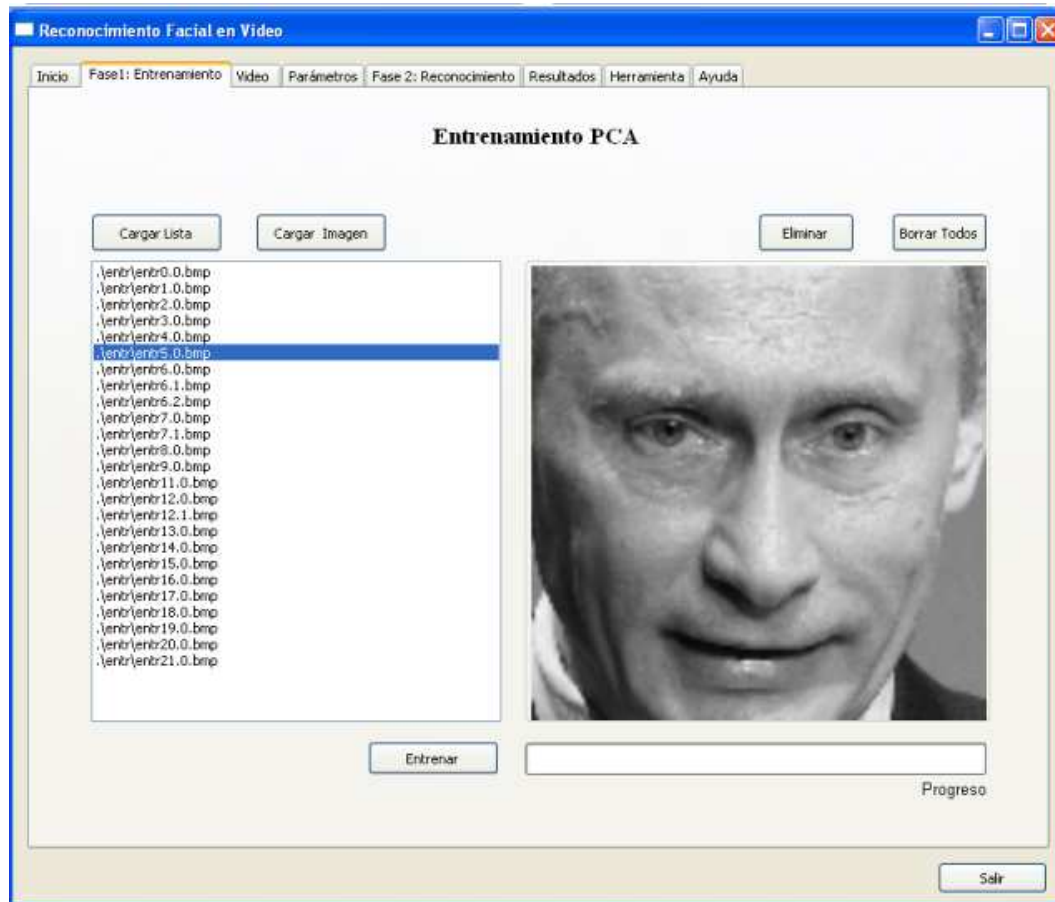


Figura 3: Interfaz de Módulo de Entrenamiento.

Sobre cada una de las imágenes ingresadas, el sistema automáticamente realiza un pre-procesamiento en base al algoritmo de detección de rostro Haar Cascade, el cual genera un grupo final de imágenes normalizadas en tamaño y color (solo niveles de gris) que se caracterizan por contener aisladamente solo el rostro de interés centrado en la nariz, tal como se representa en la figura 4.

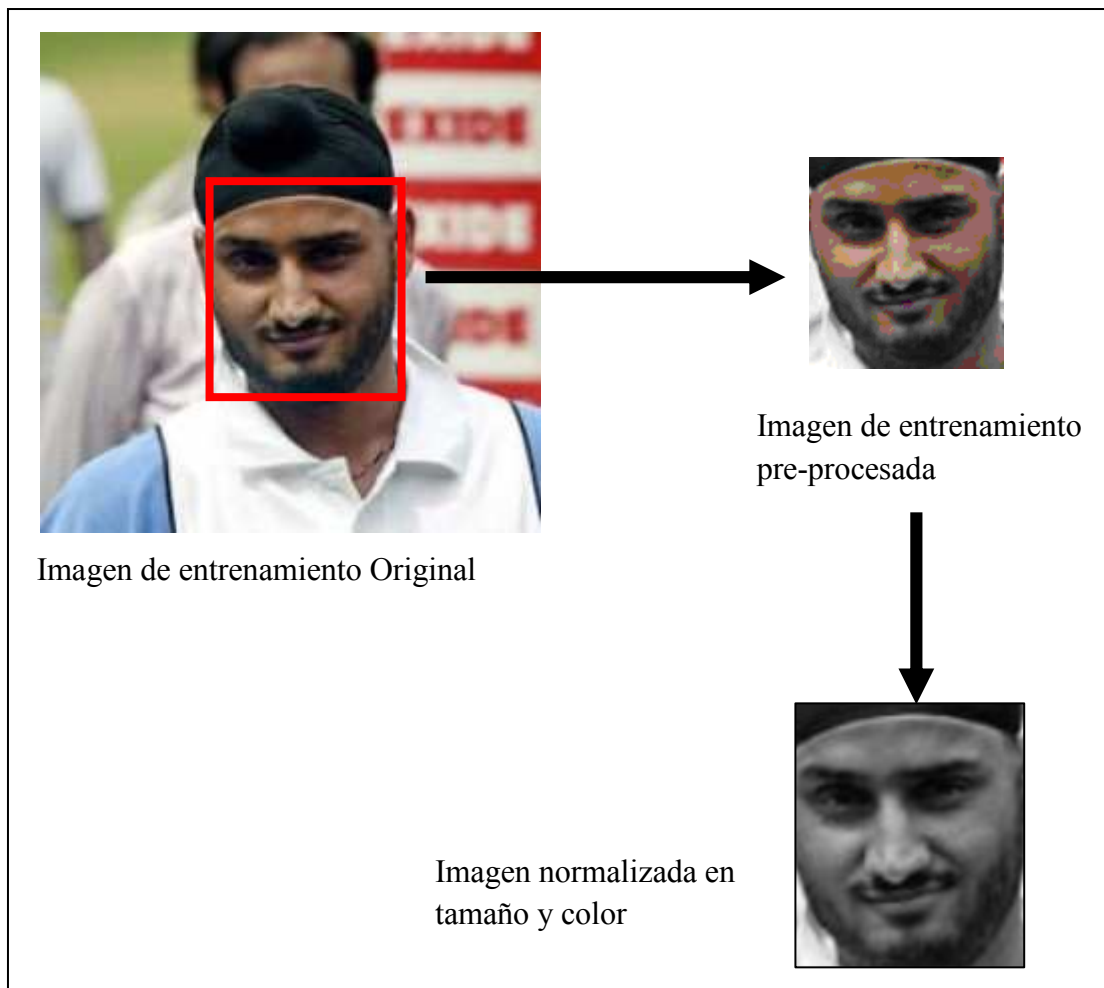


Figura 4: Pre-procesamiento de Imagen de Entrenamiento.

Tal como se muestra en la figura 4, este modulo es posible visualizar cada una de las imágenes de entrenamiento pre-procesadas, pudiendo el usuario eliminar alguna que considere de mala calidad.

Fase 1: Entrenamiento

Una vez definido el grupo de imágenes de entrenamiento, el sistema permite iniciar dicho proceso mediante el botón “entrenar” a partir del cual se ejecuta el algoritmo de entrenamiento PCA basado en Eigenface, el cual genera como resultado la base de datos de entrenamiento en formato xml llamada facedata.xml.

3.2.2. Módulo de Video

Permite realizar el ingreso del video sobre el cual se desea realizar el reconocimiento del rostro entrenado.

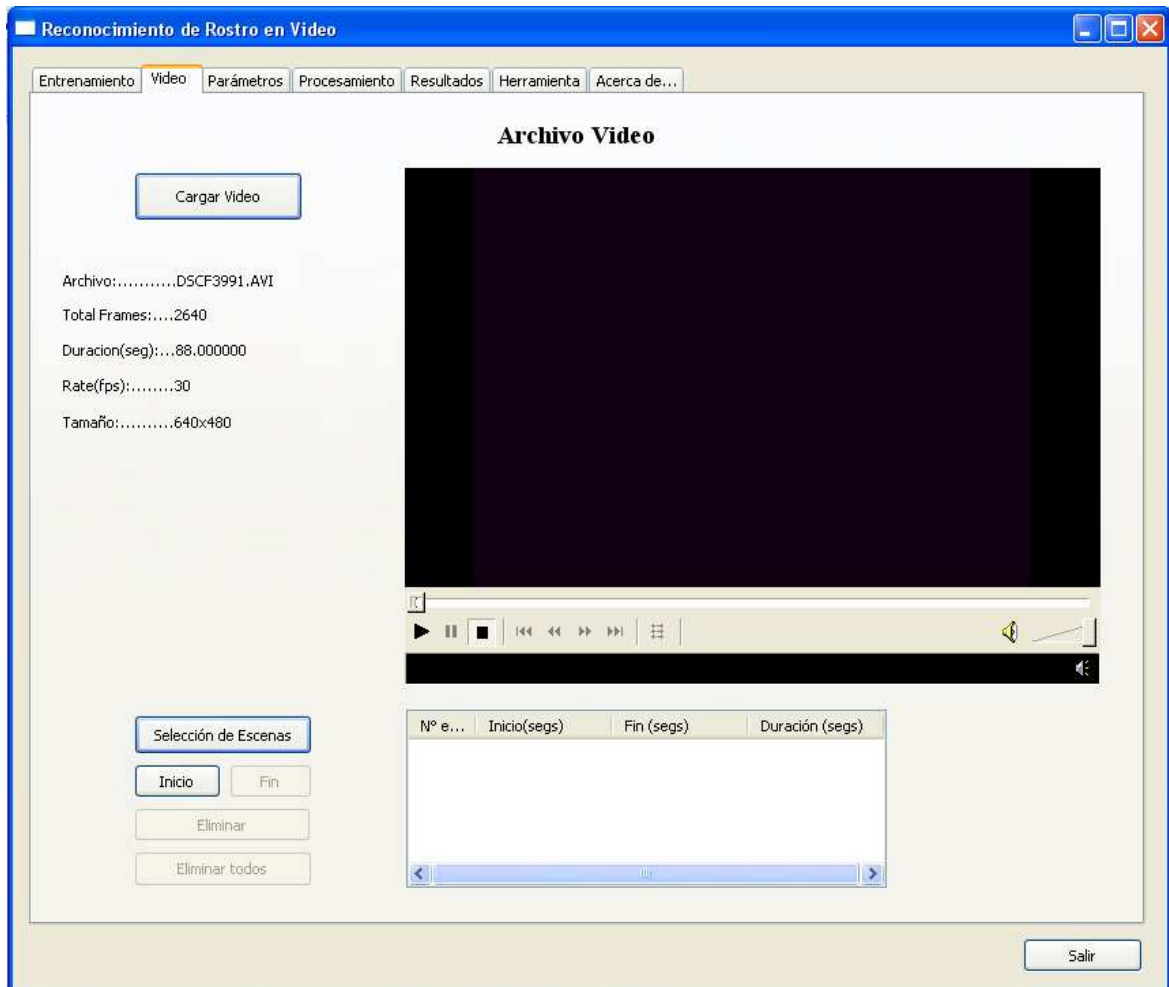


Figura 5: Interfaz de Módulo de video.

Como se observa en la figura 5, una vez ingresado el video, el sistema automáticamente despliega información general, tal como el nombre del archivo, total de cuadros que contiene, su duración en segundos, la velocidad de reproducción (rate) y el tamaño de las imágenes en pixeles.

3.2.3. Módulo Reconocimiento

Posibilita el inicio del proceso de reconocimiento en el video, desplegando una vista de cada cuadro que es procesado, realizando el despliegue de un cuadro rojo cada vez que se detecta un rostro, tal como se observa en la figura 6. También despliega datos estadísticos como lo son el total de cuadros del video y el número del cuadro en proceso. Al final de esta fase se despliega un mensaje de aviso con el tiempo del proceso.

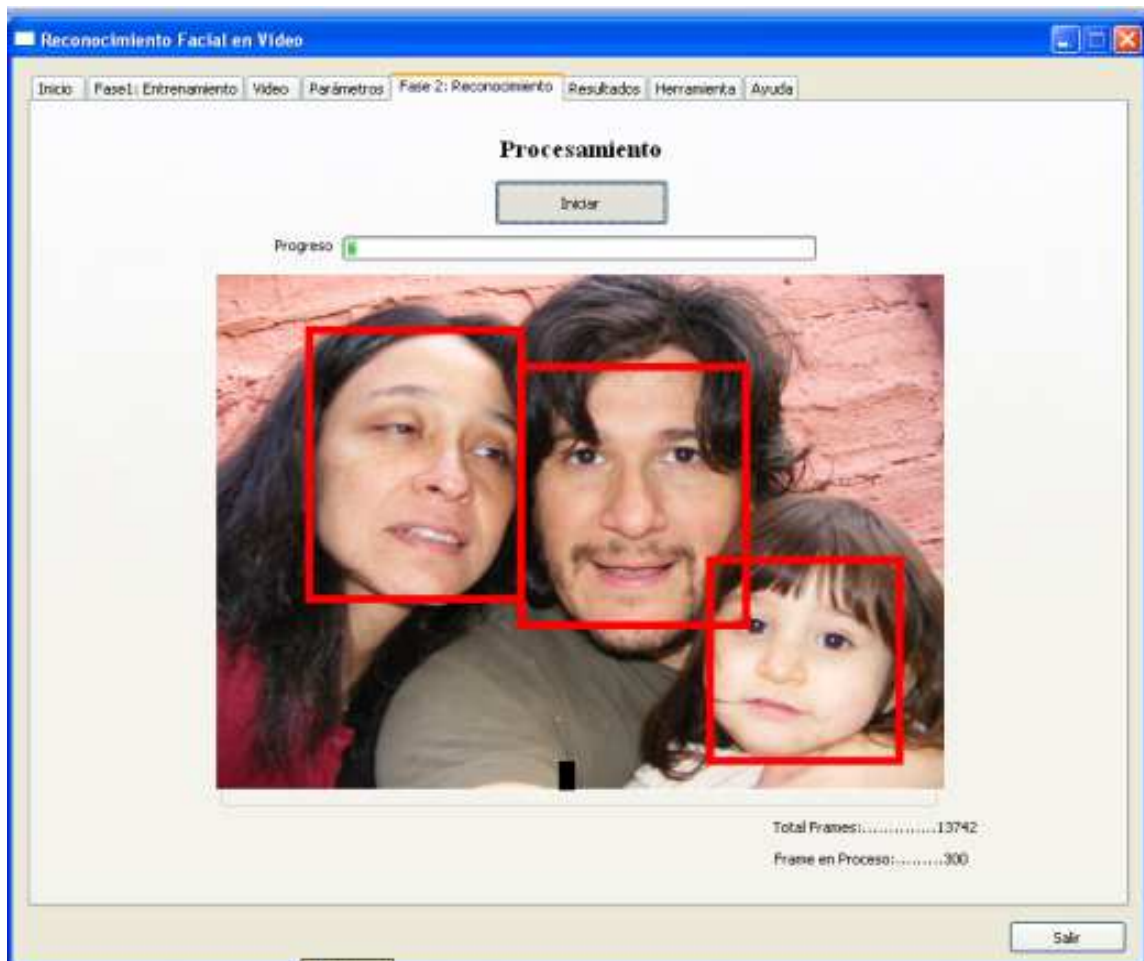


Figura 6: Módulo de Reconocimiento.

3.2.3.1. Procesamiento por Cuadro

Se realiza la descomposición secuencial del archivo de video con el fin de procesar individualmente cada imagen o cuadro que lo compone, aplicando la función de detección de rostros *Haar Cascade*. Cada rostro detectado se trata como una imagen individual que es pre-

procesada con el fin de normalizar su tamaño y color (llevados a niveles de gris), tal como se observa en la figura 7.

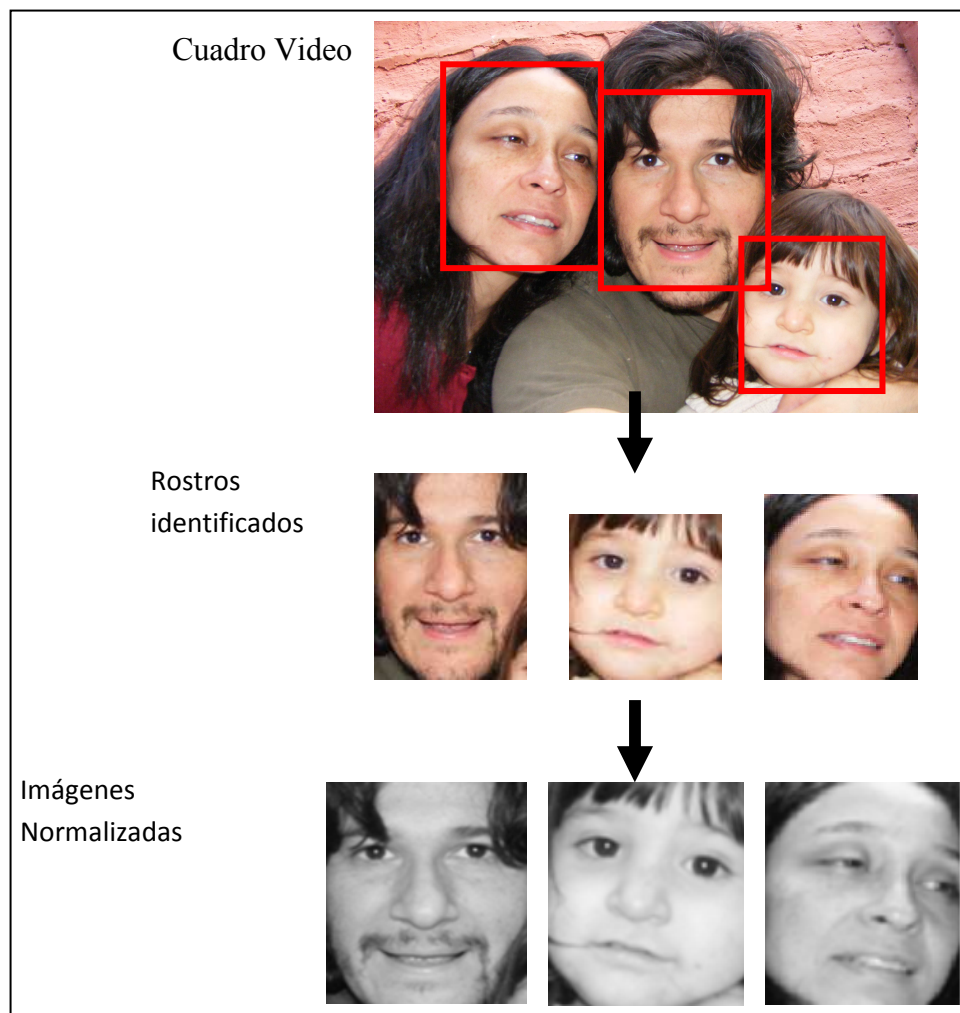


Figura 7: Pre-procesamiento de un cuadro de video para reconocimiento de rostro.

3.2.3.2. Reconocimiento

A cada imagen de rostro generada en el punto anterior, se aplica en tiempo real el algoritmo de reconocimiento que utiliza como elemento de referencia la base de datos de entrenamiento nombrada en 3.2. Este algoritmo calcula la distancia Euclideana (2.8) entre la imagen del rostro y los eigenvectores que componen el entrenamiento, información que es almacenada en forma secuencial en el archivo proceso.txt. Si esta distancia es menor o igual que el UR definido (en

módulo de configuración), el cuadro al cual pertenece el rostro en proceso se considera reconocido, y se contabiliza en la estadística de reconocimiento del sistema.

3.2.4. Módulo de Resultados

Es el encargado de entregar los resultados del proceso, para lo cual provee tres modalidades, como se observa en la figura 8: datos estadísticos resultantes del procesamiento del video, archivos de salida con el detalle de los datos obtenidos cuadro a cuadro y posibilidad de visualizar el las escenas de video en las cuales hubo (o no hubo) reconocimiento del rostro objetivo.

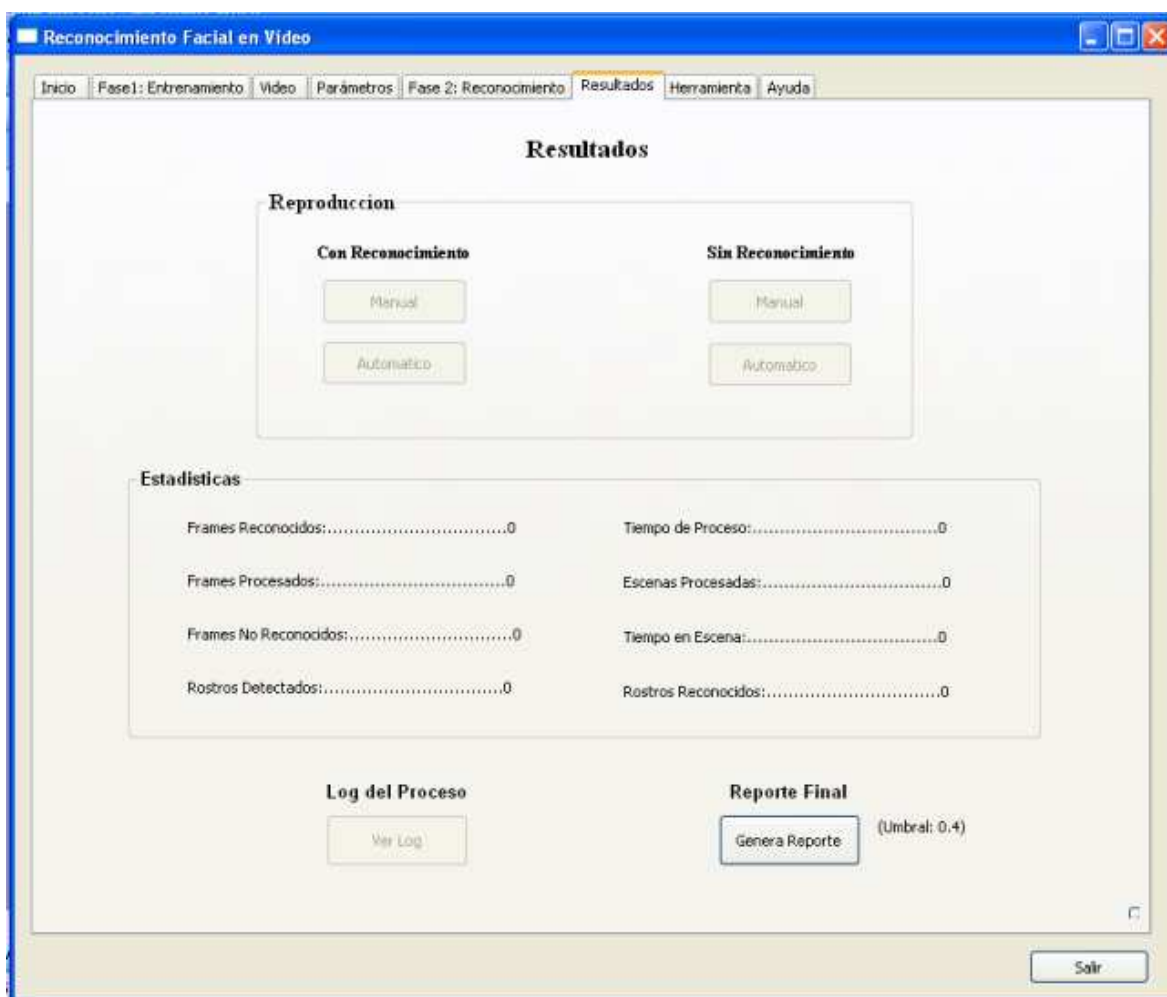


Figura 8: Interfaz de módulo de resultados.

3.2.4.1. Valores Estadísticos

Sección que presenta un resumen estadístico del proceso de reconocimiento. Los valores aquí presentados son los siguientes:

- *Cuadros Procesados*: Corresponde al total de cuadros que contiene el video.
- *Tiempo de proceso*: Tiempo transcurrido desde el procesamiento del primer cuadro hasta el último.
- *Rostros detectados*: Número total de rostros detectados en el video procesado.
- *Rostros reconocidos*: Número total de rostros reconocidos en el video procesado.
- *Cuadros Reconocidos*: Número total de cuadros que contienen el rostro reconocido.
- *Tiempo en escena*: Corresponde al tiempo en escena del rostro objetivo, a partir del reconocimiento realizado por el sistema.

3.2.4.2. Archivos de Salida

Sección que permite el despliegue de los archivos de salida del sistema:

- *Proceso.txt*: Corresponde al log del proceso que contiene el detalle del proceso de cada cuadro, en cuanto a cantidad de rostro detectados y distancia de cada rostro respecto al entrenamiento. Para conocer el detalle de la información de este archivo, ver Anexo 5.
- *Consolidado.txt*: Generado a partir del proceso de contrastación entre el archivo proceso.txt y el archivo GT.txt. Para conocer el detalle de la información de este archivo, ver Anexo 5.

3.2.4.3. Visualización de Escenas con Reconocimiento

Sección que permite la visualización del video solo en las secciones donde se produjo reconocimiento de rostro. Para ello se dispone de un botón que despliega una pantalla del tipo ‘pop up’ con dichas escenas. También se presenta la posibilidad de hacer una visualización cuadro a cuadro en forma manual.

Las mismas operaciones antes descritas es posible realizarlas para las secciones del video en las cuales no se produjo reconocimiento.

3.2.5. Módulo de Configuración

Permite la configuración de parámetros del sistema:

- Escala de para normalización de imágenes de rostro (default es 128x128 pixeles).
- Tamaño mínimo para la detección de rostro (default es 32x32 pixeles).
- Ecuilización de histograma para rostros detectados (por defecto no se aplica).
- Umbral de reconocimiento (UR)
- Activación/desactivación de archivo de log de sistema.
- Velocidad de reproducción de video dentro del módulo de resultados.
- Retornar valores por defecto del sistema.

La interfaz utilizada corresponde a la presentada en la figura 9. Para conocer el detalle de cada configuración, ver Anexo 4.

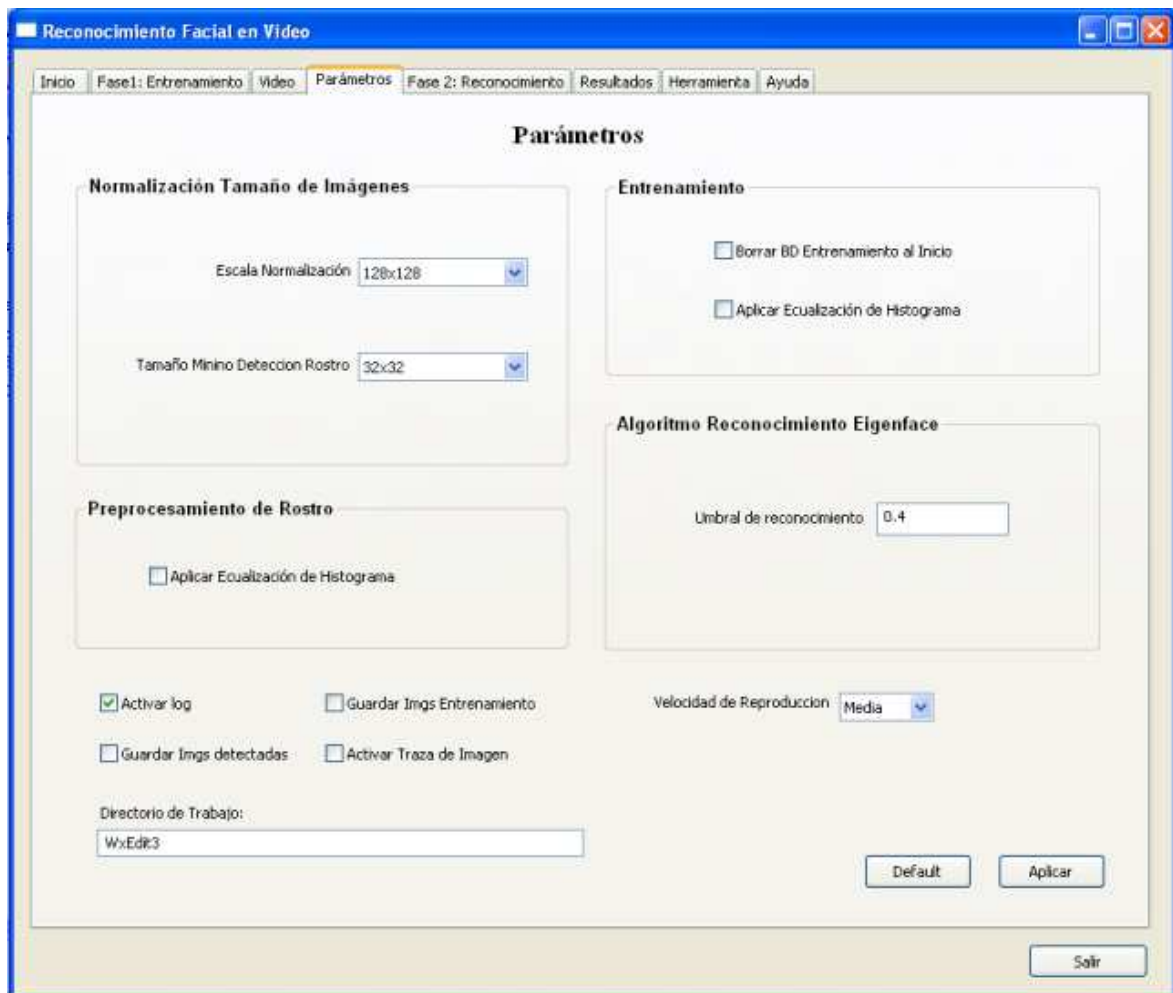


Figura 9: Interfaz de módulo de configuración.

3.2.6. Módulo Herramienta Ground Truth

Permite la generación del archivo GT.txt que contiene la información sobre la real de aparición del rostro objetivo en el video analizado. La interfaz diseñada corresponde a la presentada en la figura 10. Los detalles de operación de esta herramienta se presentan en el Anexo 4.

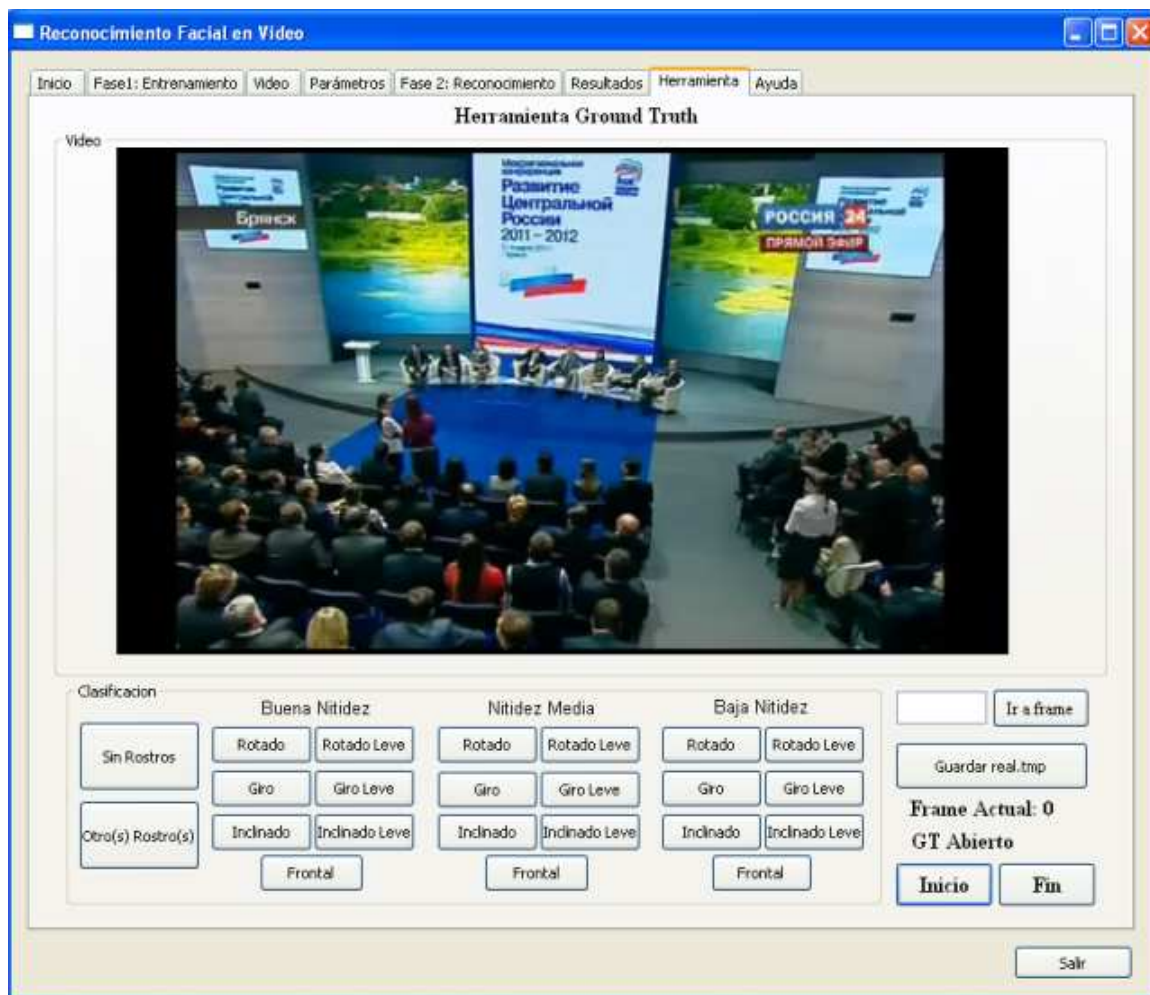


Figura 10: Interfaz herramienta Ground Truth.

3.3. Contrastación de Resultados con Referencia ‘Ground Truth’

En el módulo de Resultados se implementó un botón cuya funcionalidad es la de ejecutar automáticamente la contrastación entre los datos de reconocimiento (archivo proceso.txt) y la referencia GT.txt. Dicha contrastación se realiza en forma iterativa variando el valor del UR desde -1,0 hasta +1,0 con pasos incrementales de 0,01 (200 valores posibles).

Para cada valor de UR se obtienen los valores estadísticos de Verdaderos Positivos, Verdaderos Negativos, Falsos Positivos y Falsos Negativos, en base a los siguientes contextos:

Tipo de Resultado	Descripción
Verdaderos Positivos (VP)	Rostro SI está en cuadro y SI es identificado por el programa
Verdaderos Negativos (VN)	Rostro NO está en cuadro y NO es identificado por el programa
Falsos Positivos (FP)	Rostro NO está en cuadro y SI es identificado por el programa
Falsos Negativos (FN)	Rostro SI está en cuadro y NO es identificado por el programa

Tabla 2: Interpretación de VP, VN, FP y FN.

Los resultados de VP, VN, FP, FN obtenidos para cada valor de UR se almacenan en el archivo llamado Consolidado.txt en el cual se incluyen automáticamente los valores de Sensibilidad y Especificidad [19] en base a los cuales se grafica la curva ROC (acrónimo de Receiver Operating Characteristic) [1].

En la siguiente figura se describe secuencialmente el proceso de contrastación hasta llegar a la curva ROC y valor AUC correspondiente, los cuales finalmente caracterizarán el modelo de reconocimiento en base al entrenamiento realizado:

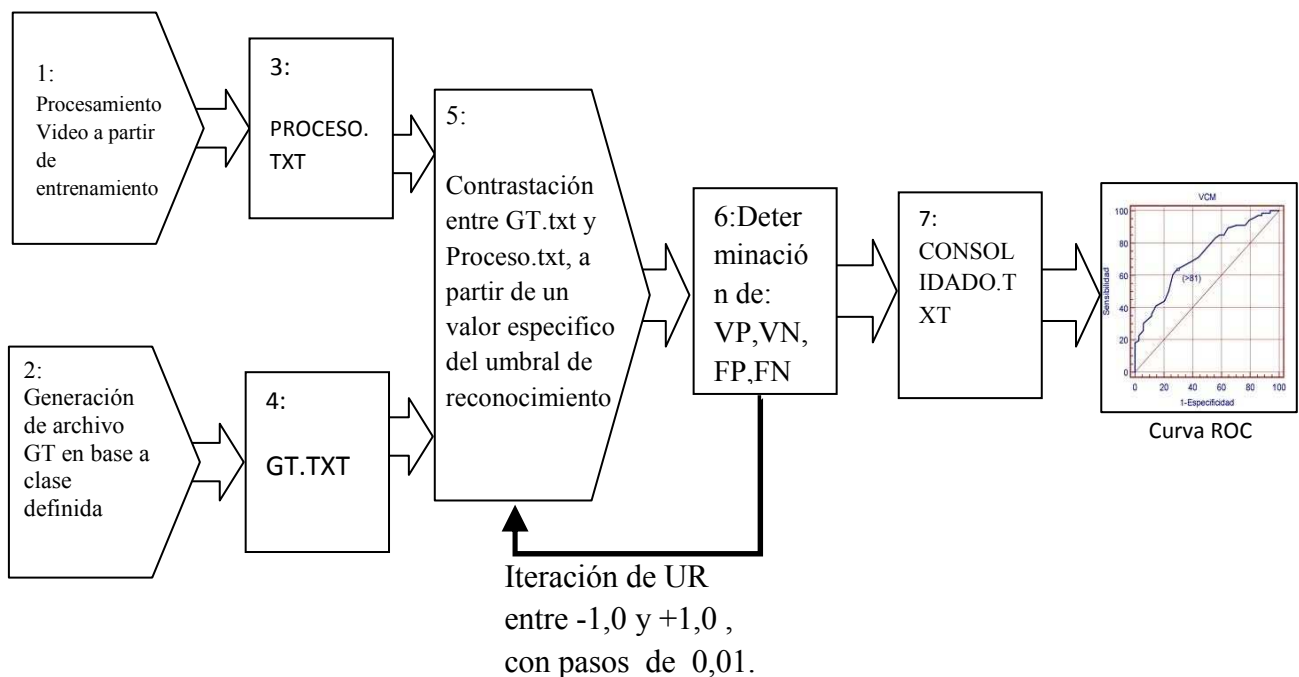


Figura 11: Diagrama secuencial del procesamiento de información de una prueba.

Para la generación de gráficos de las curvas ROC se utilizó herramienta GRAPH Versión 4.4. Para el cálculo del índice AUC (área bajo la curva) se utilizó una funcionalidad provista por la misma herramienta, utilizando el método del promedio de la serie de aproximaciones trapezoidales [19].

Capítulo 4

4. Pruebas y Resultados

En este capítulo se describen las pruebas ejecutadas para verificar la validez del sistema desarrollado. También se detalla el procesamiento de datos realizado en cada prueba, para finalmente presentar los resultados obtenidos en la modalidad de gráficos ROC. Finalmente se presenta una discusión de los resultados obtenidos.

4.1. Datos de prueba

4.1.1. Selección de Datos de Prueba

Se determinó el uso de rostros de prueba en base a imágenes recolectadas desde Internet, para lo cual se seleccionaron tres personajes con una gran variedad de fotografías disponibles, sin restricciones de derechos de autor:

- 100 imágenes de Barak Obama (Presidente de EEUU), en adelante Rostro A.
- 100 imágenes de Vladimir Putin (Presidente de Rusia), en adelante Rostro B.
- 100 imágenes de Nicolás Sarkozy (Presidente de Francia), en adelante Rostro C.

Las imágenes recolectadas contienen distinta luminosidad, pose y gesto, incluyendo las siguientes restricciones: imagen frontal, tamaño del rostro circunscrito, a lo menos, en 128x128 píxeles, con ángulo de inclinación, rotación y giro no superior a ± 20 grados [28]. En la siguiente figura se grafican los conceptos de giro en torno al eje vertical, inclinación en torno al eje horizontal y rotación en el plano frontal del rostro.

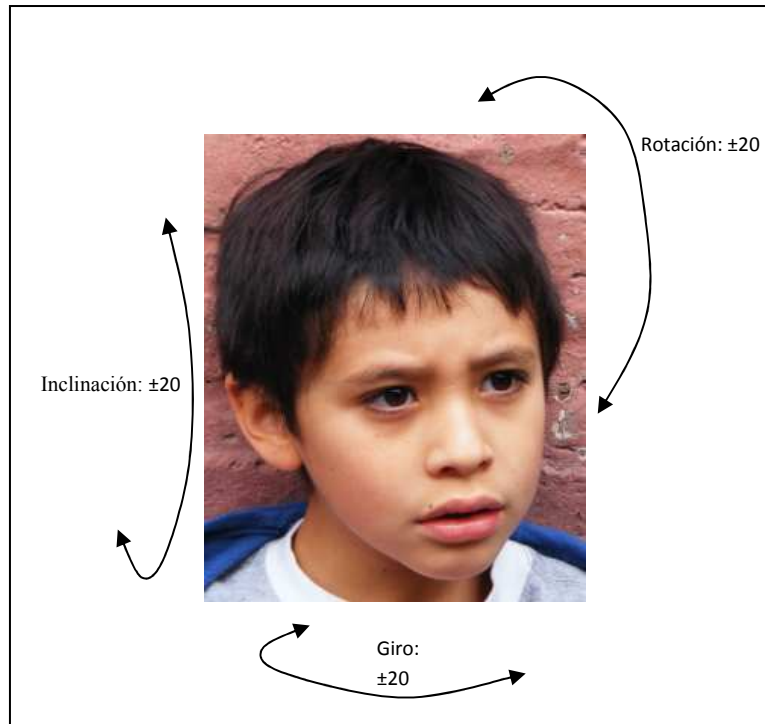


Figura 12: Ejes de movimiento posibles de un rostro frontal.

4.1.2. Selección de Videos de Prueba

Se seleccionaron 3 videos de alta definición (1920×1080 píxeles) desde Internet en los cuales participan los rostros de prueba mencionados, en condiciones de pose y gesto diverso, incluyendo la participación de rostros adicionales:

- Video A: Entrevista televisiva a Barak Obama que incluye otros cinco personajes, con una duración de 8 minutos.
- Video B: Foro televisivo en el cual participa Vladimir Putin junto a otros 15 personajes, con una duración de 6 minutos.
- Video C: Entrevista televisiva a Nicolás Sarkozy junto a otros 4 personajes, con una duración de 7 minutos.

Adicionalmente se construyó una segunda base con 100 imágenes para cada rostro de prueba, utilizando imágenes extraídas de cada uno de los videos A, B y C.

4.1.3. Sets de Prueba

A continuación se presentan los grupos de imágenes y video utilizados para la ejecución de pruebas.

4.1.3.1. Set de Prueba A

Está compuesto por Video A (figura 13), 100 imágenes del rostro A obtenidas desde Internet, en adelante, del tipo ‘on-wild’ (figura 14) y 100 imágenes del rostro A extraídas desde el mismo video A (figura 15).



Figura 13: Muestra de video de prueba A (Obama).



Figura 14: Muestra imágenes del rostro A on-wild.



Figura 15: Muestra imágenes del rostro A extraídas desde el video.

4.1.3.2. Set de Prueba B

Esta compuesto por Video B (figura 16), 100 imágenes del rostro B ‘on-wild’ (figura 17) y 100 imágenes del rostro B extraídas desde el mismo video B (figura 18).



Figura 16: Muestra de video de prueba B (Putin).



Figura 17: Muestra imágenes del rostro B on-wild.



Figura 18: Muestra de imágenes del rostro B obtenidas del video.

4.1.3.3. Set de Prueba C

Esta compuesto por Video C (figura 19), 100 imágenes del rostro C on-wild (figura 20) y 100 imágenes del rostro B extraídas desde el mismo video B (figura 21).



Figura 19: Muestra de video de prueba C (Sarkozy).

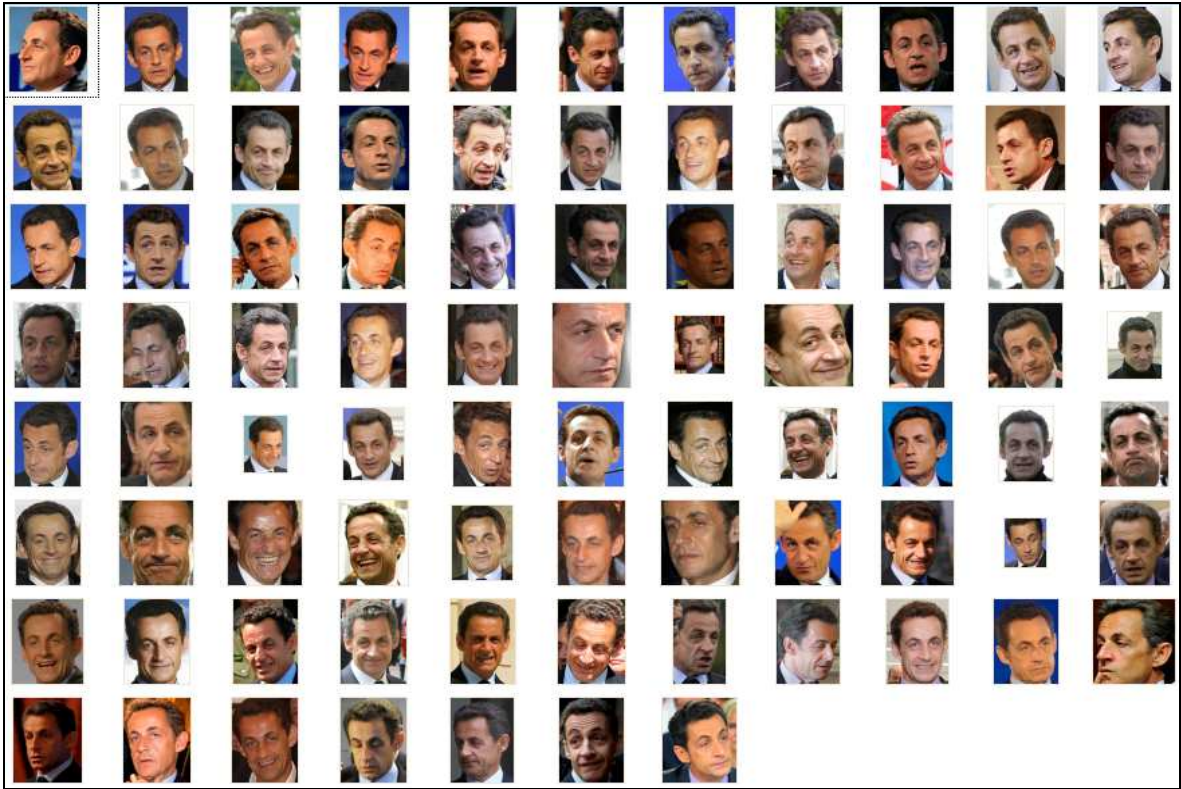


Figura 20: Muestra de imágenes del rostro C on-wild.



Figura 21: Muestra de imágenes del rostro C obtenidas del video.

4.2. Pruebas

Se realizaron pruebas para cada rostro (A, B y C) considerando un número de imágenes de entrenamiento (en adelante, NIE) correspondiente a 20, 30, 40 y 100, en dos escenarios:

Pruebas Tipo 1: NIE igual a 20, 30, 40 y 100, utilizando las fotografías extraídas del video de prueba del rostro correspondiente.

Pruebas Tipo 2: NIE igual a 20, 30, 40 y 100, utilizando fotografías del tipo *On-Wild*.

Para las agrupaciones de 20, 30, 40 y 100 imágenes, se considera prioritariamente las imágenes cualitativamente (luminosidad, pose y gesto) más cercanas a las apariciones de rostro de prueba en el video.

Las tablas 3, 4 y 5, resumen los 8 tests realizados por cada rostro de prueba:

Pruebas Rostro A				
Tipo de Prueba	NIE			
	20	30	40	100
Tipo 1	P1.1	P1.2	P1.3	P1.4
Tipo 2	P1.5	P1.6	P1.7	P1.8

Tabla 3: Pruebas Rostro A

Pruebas Rostro B				
Tipo de Prueba	NIE			
	20	30	40	100
Tipo 1	P2.1	P2.2	P2.3	P2.4
Tipo 2	P2.5	P2.6	P2.7	P2.8

Tabla 4: Pruebas Rostro B

Pruebas Rostro C				
Tipo de Prueba	NIE			
	20	30	40	100
Tipo 1	P3.1	P3.2	P3.3	P3.4
Tipo 2	P3.5	P3.6	P3.7	P3.8

Tabla 5: Pruebas Rostro C.

4.3. Evaluación de las Pruebas

La evaluación de cada prueba se realiza mediante el uso de curvas ROC generadas a partir de la metodología planteada en 3.8. Para cada curva ROC se obtiene el índice AUC con el cual se determina la calificación del sistema en cuanto a su capacidad de reconocimiento, según el criterio de intervalos [19] para los valores de AUC presentado en la Tabla 6.

Rango AUC	Calificación
[0.5 a 0.6):	Test malo
[0.6 a 0.75):	Test regular
[0.75 a 0.9):	Test bueno
[0.9 a 0.97):	Test muy bueno
[0.97 a 1):	Test excelente

Tabla 6: Criterio de intervalos para valores de AUC.

4.4. Resultados

A continuación se presentan los resultados obtenidos a partir de las pruebas ejecutadas con los sets A, B y C:

4.4.1. Resultados Set de Prueba Rostro A

4.4.1.1. Gráficos ROC Pruebas Tipo 1

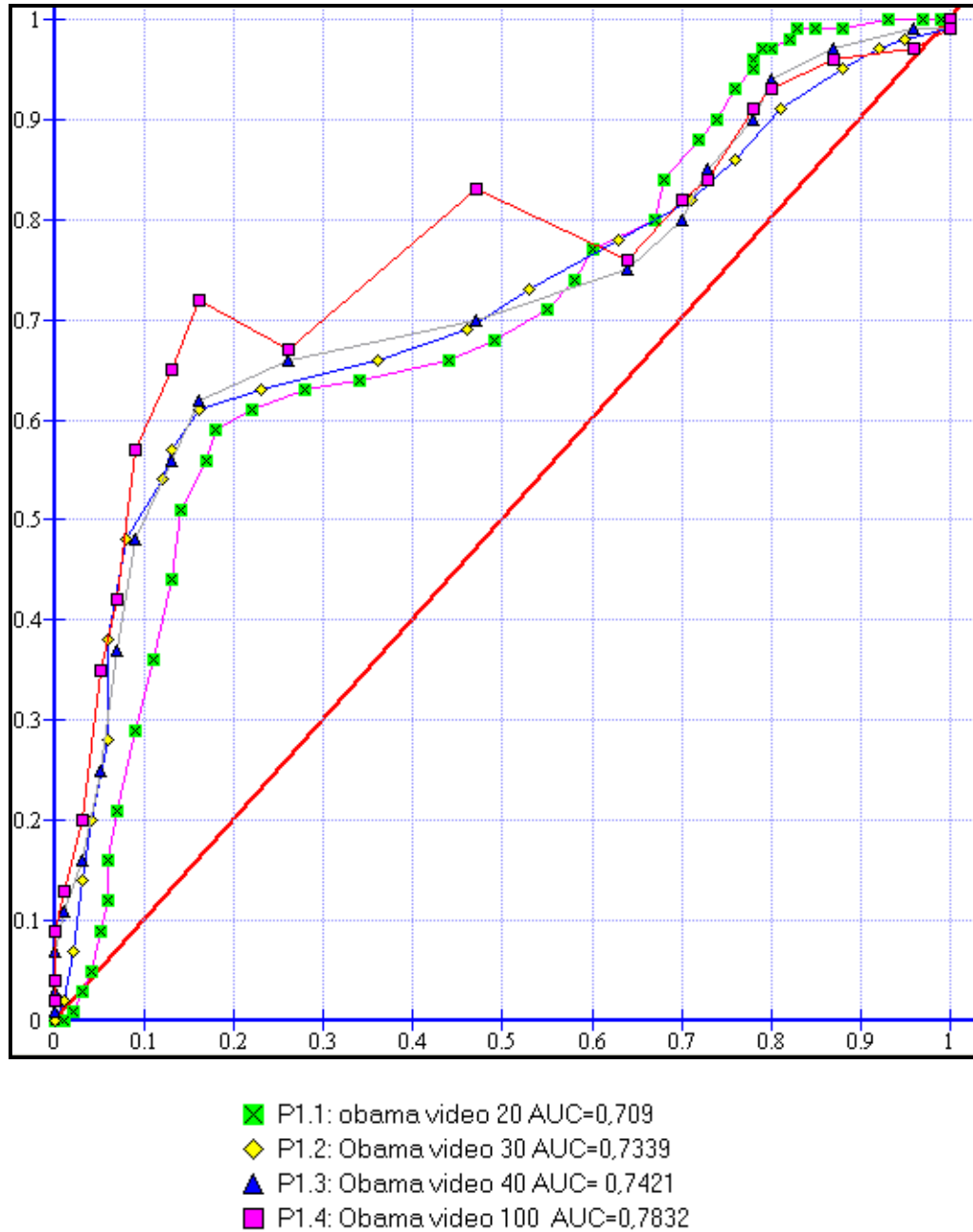


Figura 22: Curvas ROC pruebas P1.1, P1.2, P1.3, P1.4

4.4.1.2. Gráficos ROC Pruebas Tipo 2

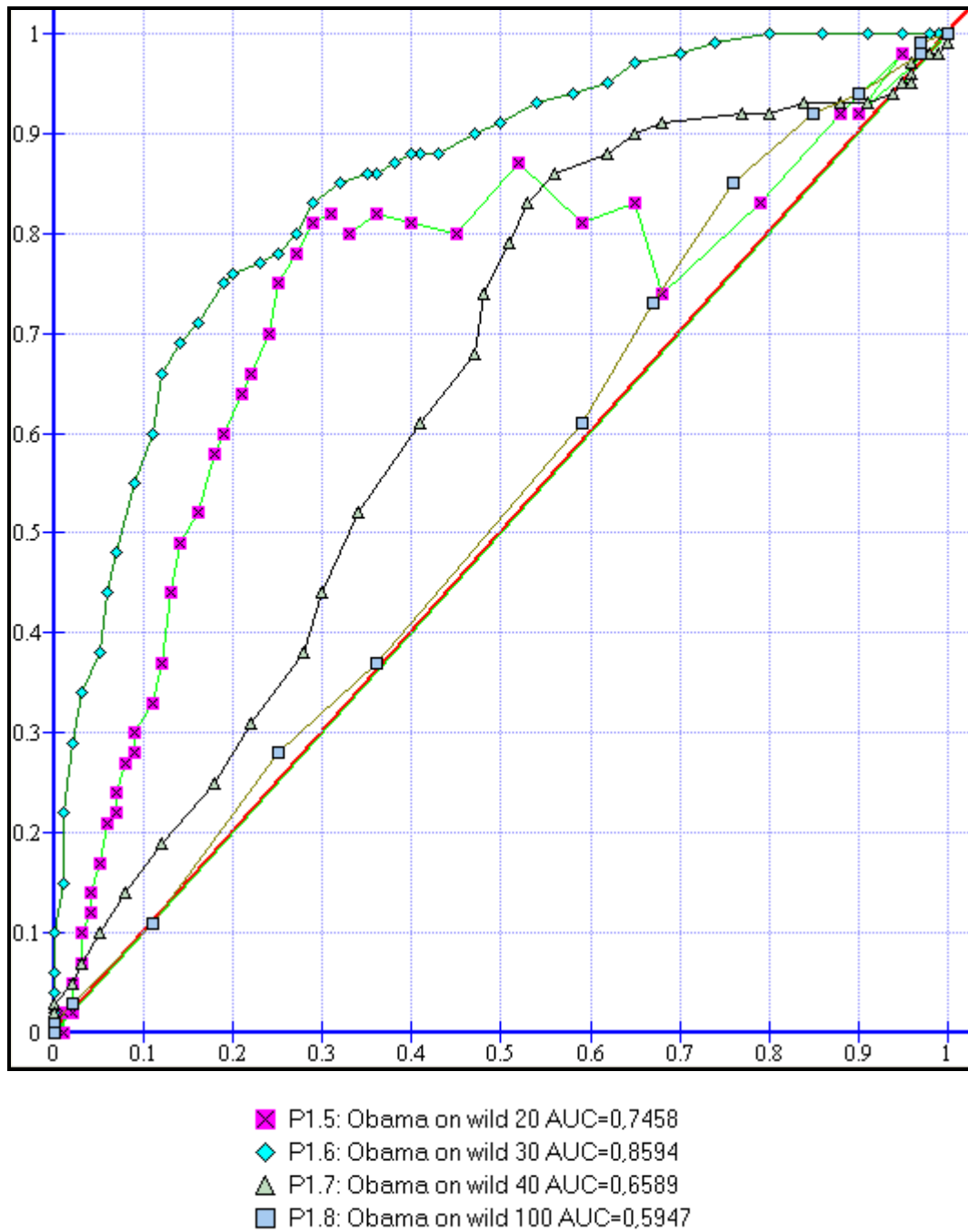


Figura 23: Curvas ROC pruebas P1.5, P1.6, P1.7, P1.8

ta4.4.2. Resultados Set de Prueba Rostro B

4.4.2.1. Gráficos ROC Pruebas Tipo 1

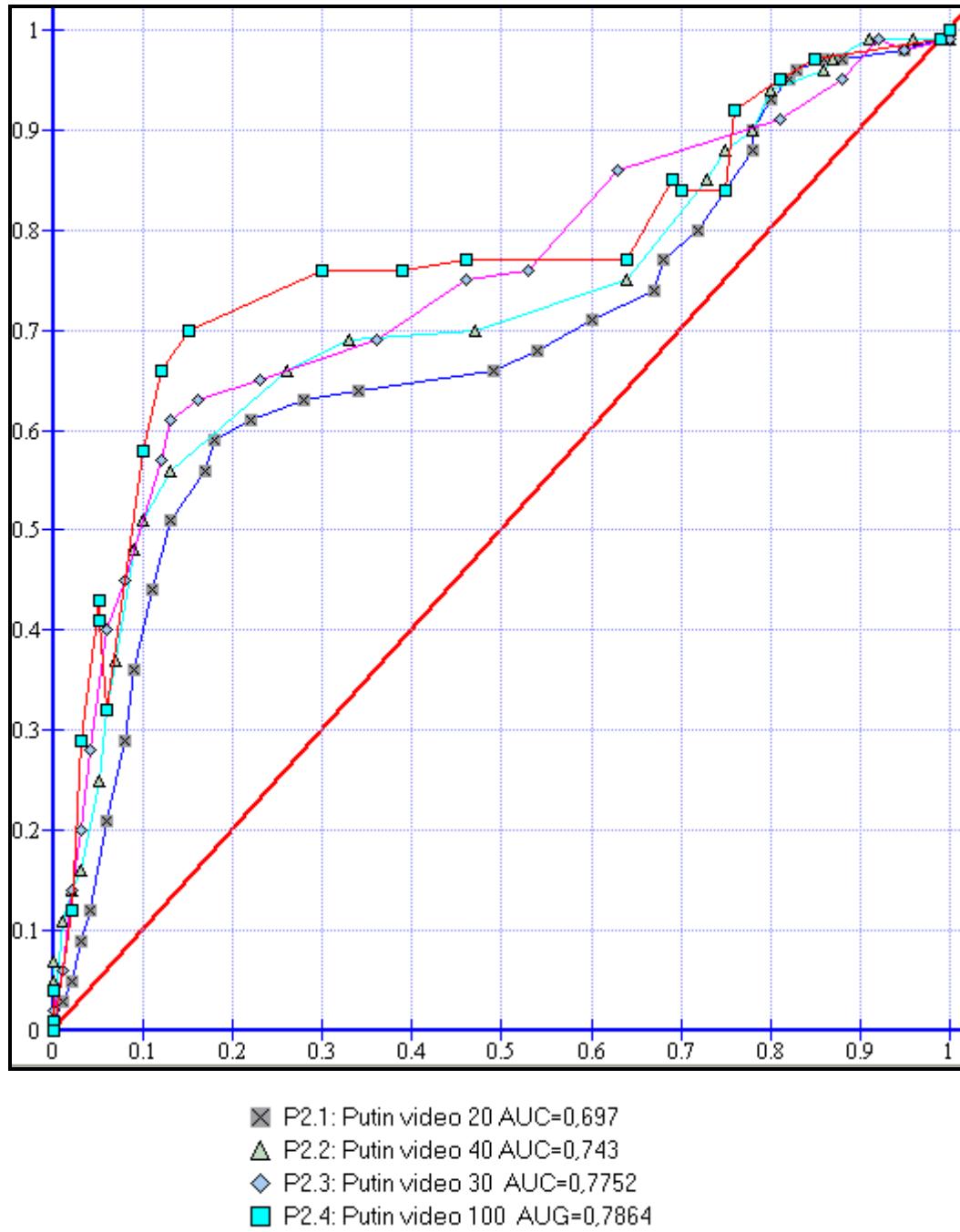


Figura 24: Curvas ROC pruebas P2.1, P2.2, P2.3, P2.4

4.4.2.2. Gráficos ROC Pruebas Tipo 2

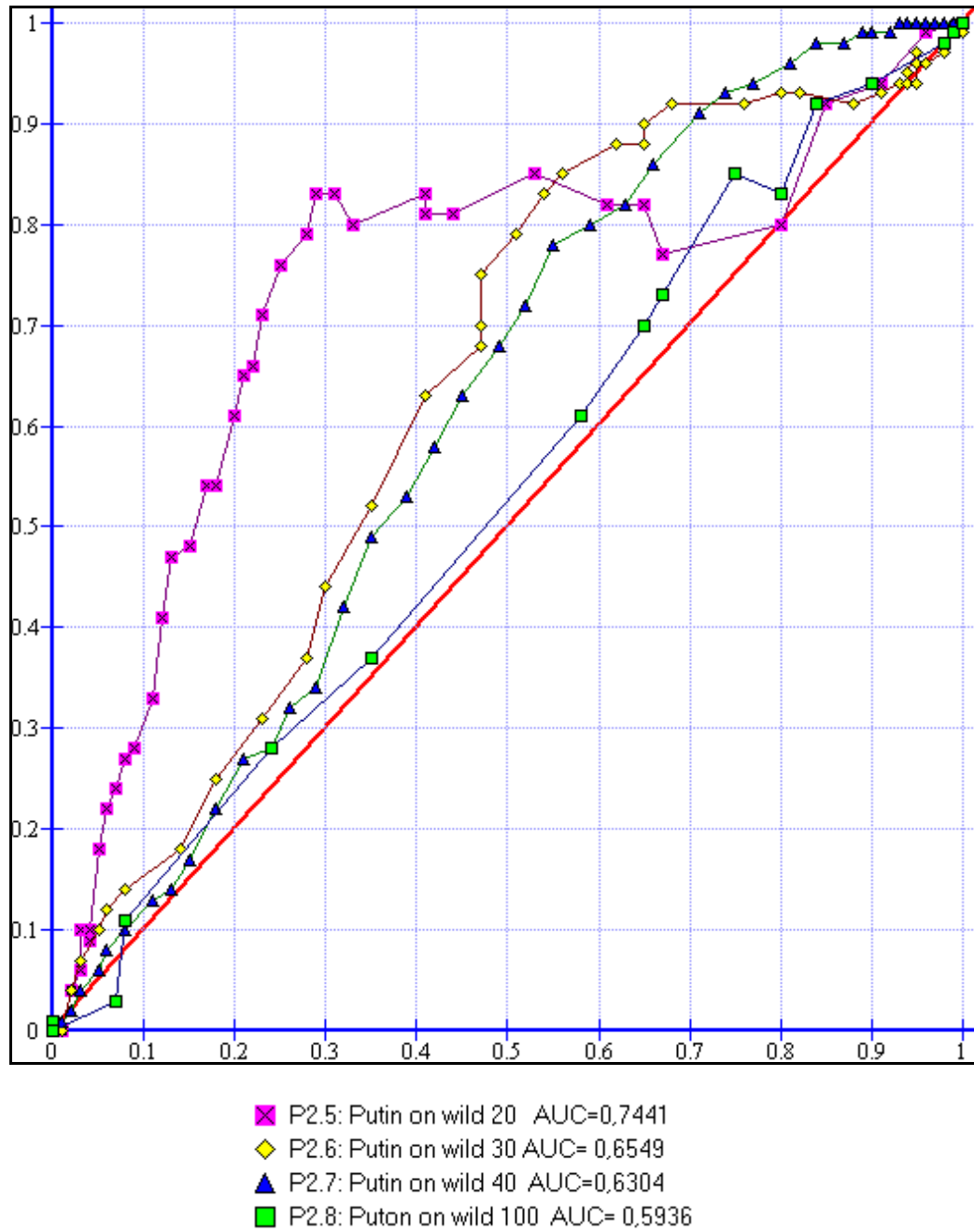


Figura 25: Curvas ROC pruebas P2.5, P2.6, P2.6, P2.7

4.4.3. Resultados Set de Prueba Rostro C

4.4.3.1. Gráficos ROC Pruebas Tipo 1

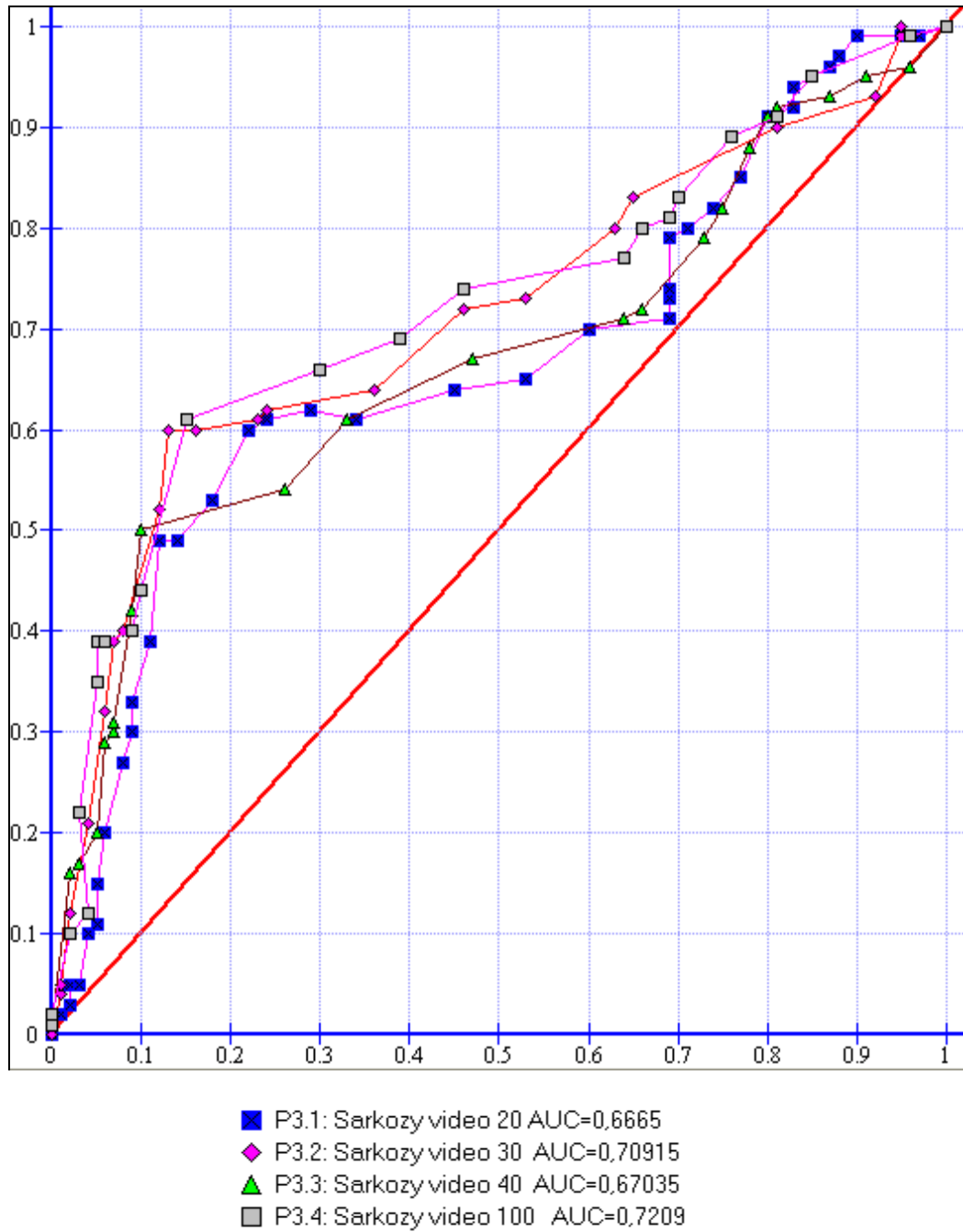


Figura 26: Curvas ROC pruebas P3.1, P3.2, P3.3, P3.4

4.4.3.2. Gráficos ROC Pruebas Tipo 2

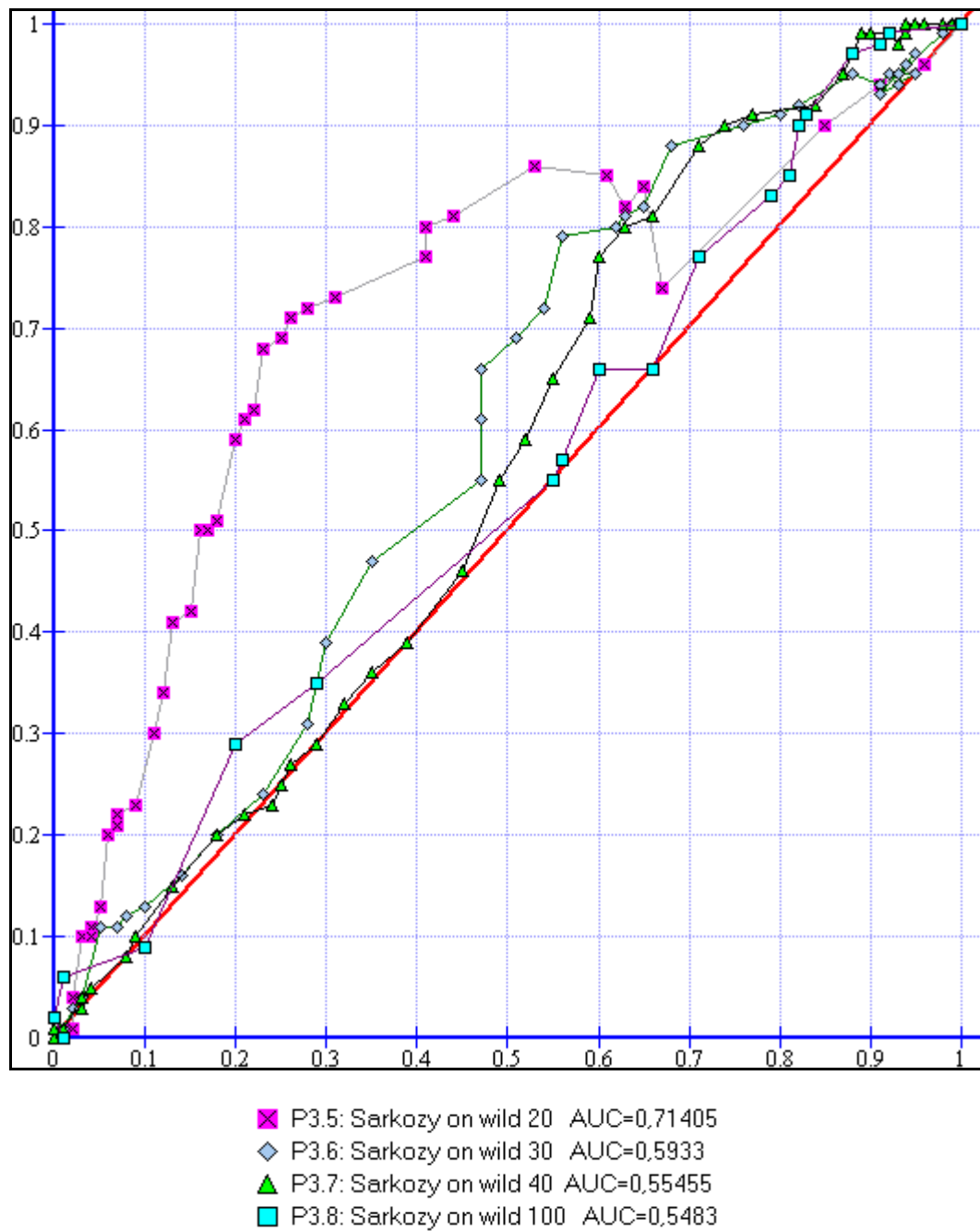


Figura 27: Curvas ROC pruebas P3.5, P3.6, P3.7, P3.8

4.4.4. Índice AUC

En las siguientes figuras 28, 29 y 30, se representa el comportamiento del índice AUC versus NIE, considerando las pruebas Tipo 1 y 2, para cada set de pruebas.

4.4.4.1 Set de Prueba A

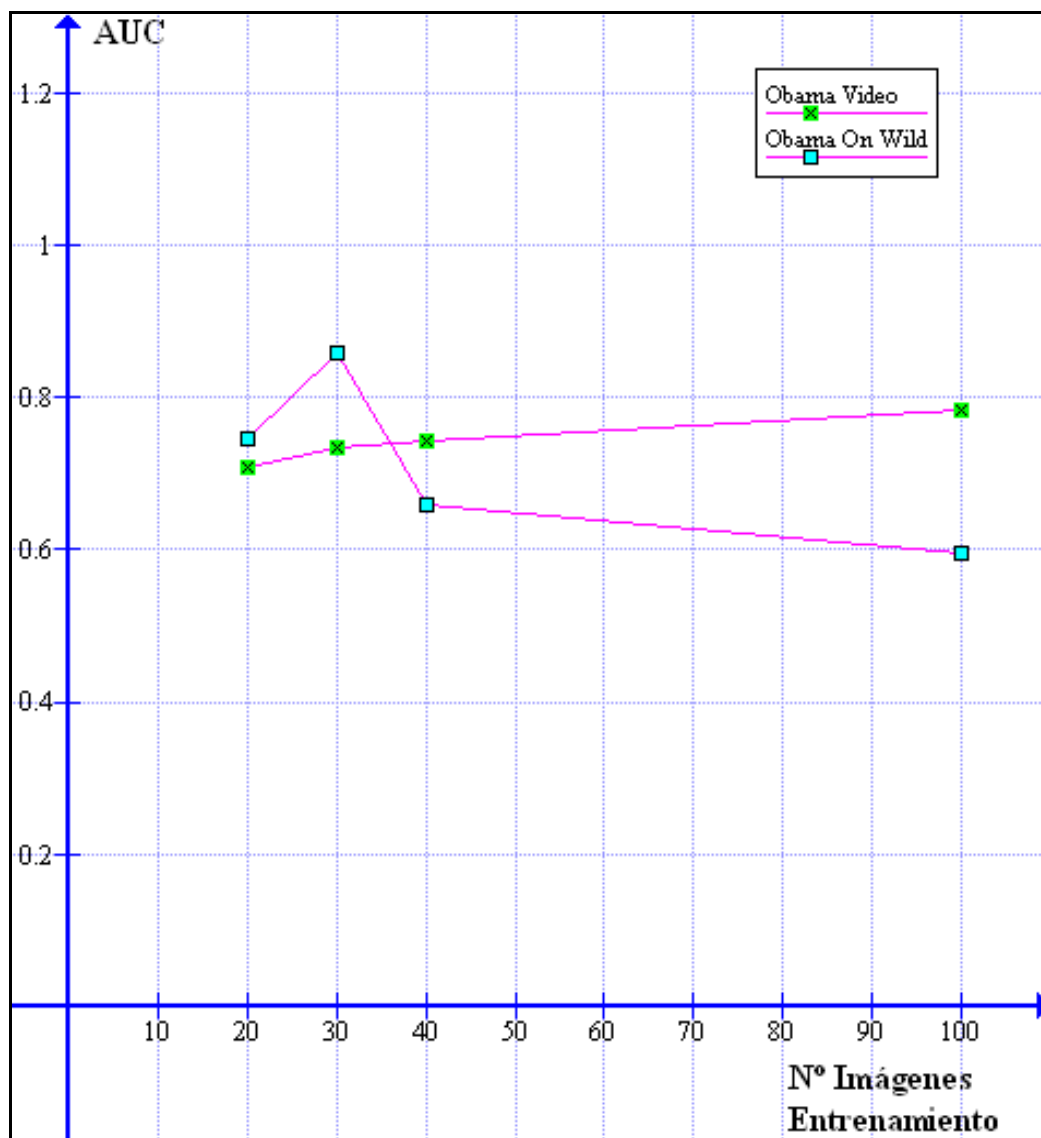


Figura 28: Comportamiento AUC set de pruebas A.

4.4.4.2. Set de Prueba B

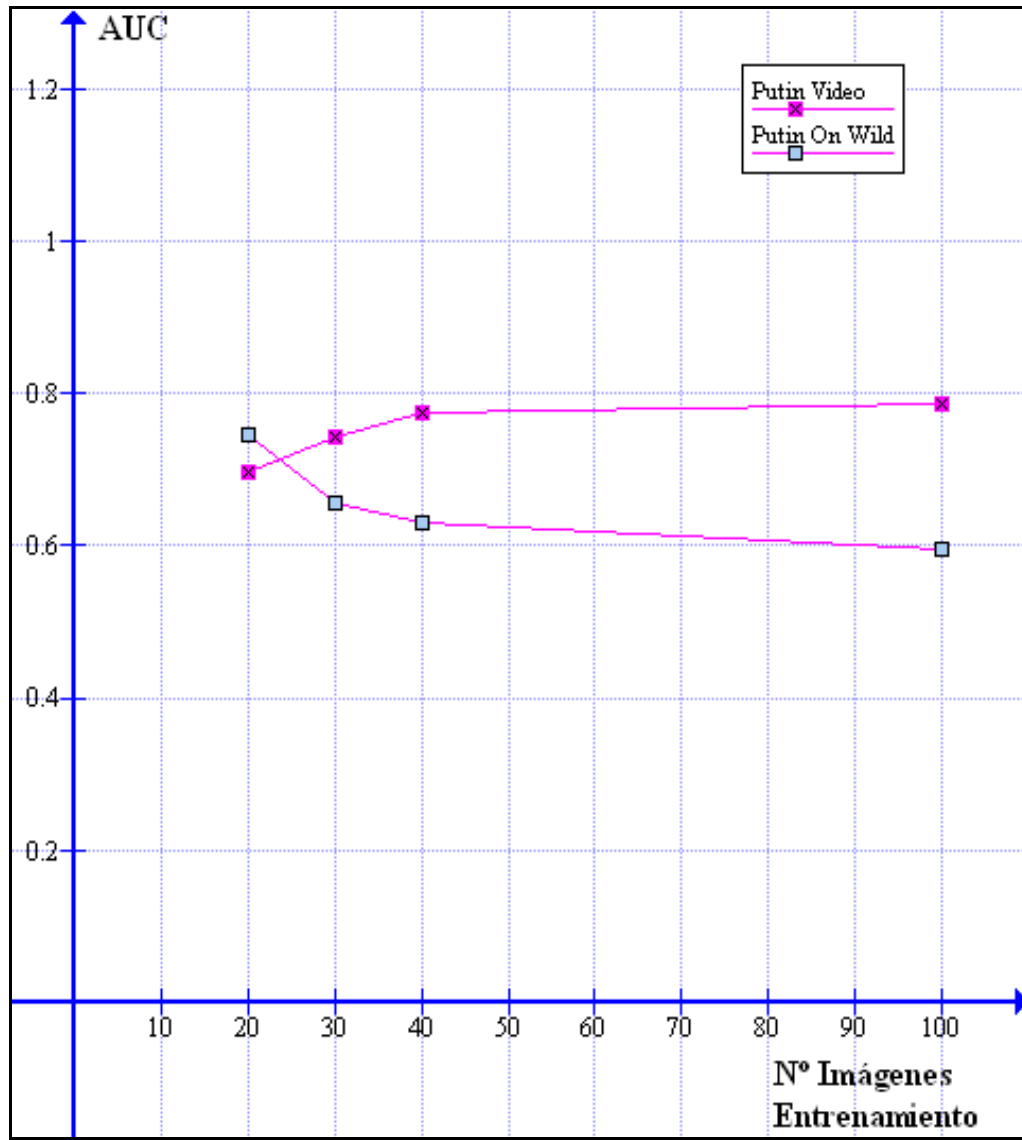


Figura 29: Comportamiento AUC set de prueba B.

4.4.4.3. Set de Prueba C

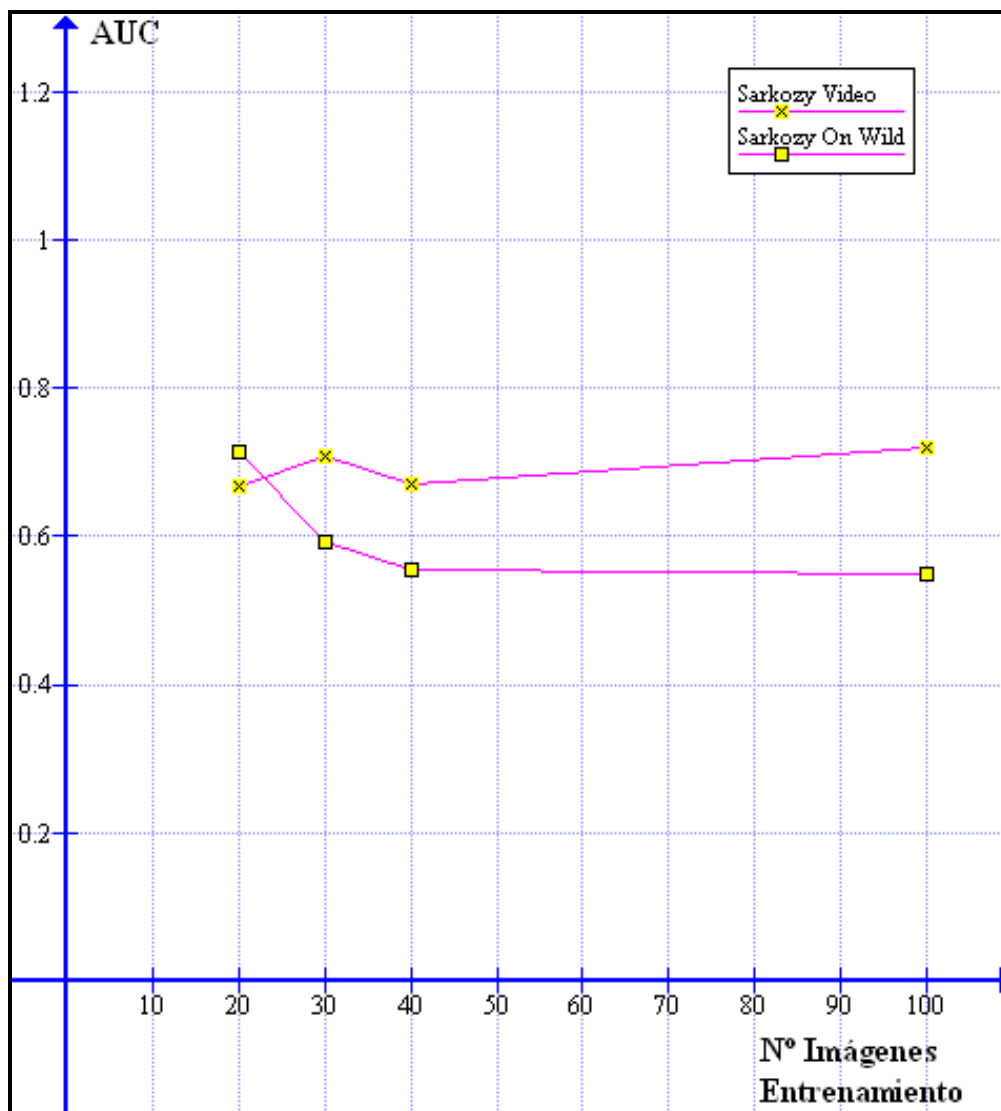


Figura 30: Comportamiento AUC set de prueba B.

4.5. Discusión de Resultados

4.5.1. Pruebas Tipo 1

En las curva ROC de pruebas Tipo1 del rostro A (Fig. 22) se observa la mejor respuesta del sistema para el caso NIE igual a 100 (prueba P1.4), donde si se acepta una tasa de falsos positivos de hasta un 15%, se observa una tasa de aciertos del orden de un 73%.

En las siguientes curvas donde el entrenamiento se realiza con 40, 30 y 20 imágenes (pruebas P1.1, P1.2 y P1.3), se observa una degradación en la calidad del sistema, obteniendo curvas ROC

mas planas y cercanas a las diagonal, alcanzando para el caso NIE=20 (prueba P1.1) la performance más baja.

Para los rostros de prueba B y C se observan curvas ROC (Fig. 24 y Fig. 26) un comportamiento similar la del rostro A, presentando mayor performance para el caso de NIE igual a 100 (pruebas P2.4 y P3.4 respectivamente) y menor para el caso de NIE igual a 20 (pruebas P2.1 y P3.1 respectivamente).

A partir de lo anterior y se puede plantear que para este tipo de pruebas, al realizar un entrenamiento con un mayor numero de imágenes del mismo video, se obtiene una mayor capacidad de reconocimiento, lo cual tiene sentido, ya que dichas imágenes contienen una naturaleza similar al video en cuanto a luminosidad, pose y gestos.

4.5.2. Pruebas Tipo 2

En las curvas ROC de pruebas Tipo 2 del rostro A (Fig. 23) se observa la mejor respuesta del sistema para NIE igual a 30 (prueba P1.6), donde si se acepta una tasa de falsos positivos de hasta un 15%, se observa una tasa de aciertos aproximadamente de un 70%. El resultado decae drásticamente en las siguientes curvas con NIE igual a 40 y 100 (pruebas P1.7 y P1.8), siendo este ultimo caso el que presenta una curva ROC casi plana y en línea con la diagonal representando un sistema prácticamente sin capacidad de discriminación [1].

Para los rostros de prueba B y C se observan curvas ROC (Fig. 25 y Fig. 27) que representan una mejor respuesta del sistema para NIE igual a 20 (pruebas P2.5 y P3.5 respectivamente), donde si se acepta una tasa de falsos positivos de hasta un 22%, se observa una tasa de aciertos del orden de un 68%. Para las otras curvas con valor NIE igual a 30 y 40 (pruebas P2.6, P2.7, P3.6 y P3.7 respectivamente), la performance del sistemas decae drásticamente para llegar al caso de NIE igual a 100 (pruebas P2.8 y P3.8) que presenta una curva ROC casi plana y en línea con la diagonal representando un sistema prácticamente sin capacidad de discriminación.

A partir de lo anterior se puede plantear que la selección de las 20 o 30 imágenes más cercanas a la realidad del video de prueba logra la mejor performance del sistema. Al incluir más imágenes en las cuales hay una mayor diversidad de características como gesto, posición y luminosidad se logra una degradación de la performance del sistema.

4.6.3. Índice AUC

Los gráficos del índice AUC presentados para los tres rostros de prueba (Figuras 28, 29 y 30) presentan una tendencia similar que resume el comportamiento ya descrito en 4.5.1 y 4.5.2, el cual indica que para pruebas Tipo 1 el sistema tiene mayor performance de reconocimiento mientras mayor sea el número de imágenes incluidas en el entrenamiento y para pruebas Tipo 2 el sistema tiene mayor performance en cuanto las imágenes de entrenamiento sean cualitativamente más cercanas a la realidad del rostro objetivo en el video de prueba.

En las pruebas Tipo 1, con NIE igual a 100 se obtienen los mayores valores de AUC llegando al rango entre 0,72 y 0,78 (pruebas P1.4, P2.4 y P3.4). En las pruebas Tipo 2, con NIE igual a 20 se obtienen los mayores valores de AUC llegando al rango entre 0,71 y 0,74 (pruebas P1.5, P2.5 y P3.5).

A partir de lo anterior y de acuerdo a los criterios planteados en 4.3 (Tabla 6), en las pruebas Tipo 1 el sistema presenta la mayor capacidad reconocimiento facial con un NIE igual a 100, obteniendo una calificación *buena*. En el caso de pruebas Tipo 2 el sistema presenta la mayor capacidad de reconocimiento con un NIE igual a 20, obteniendo una calificación *regular*.

Capítulo 5

5. Conclusiones y Líneas de trabajo Futuras

El sistema de reconocimiento facial desarrollado en esta memoria cumple con los siguientes objetivos planteados inicialmente:

- Satisface los criterios de diseño modular propuestos en la arquitectura inicial.
- A partir de un conjunto de imágenes de un rostro objetivo definido por un usuario, es capaz de conseguir y almacenar un entrenamiento en base a la utilización de los métodos Haar para de detección de rostro y PCA para extracción de características.
- A partir de un video de interés definido por el usuario y del entrenamiento antes mencionado, es capaz de Procesar y entregar resultados acerca el reconocimiento del rostro objetivo en el video.

Con los resultados obtenidos a partir de las pruebas realizadas, es posible plantear las siguientes conclusiones:

- En las pruebas cuyo entrenamiento fue realizado con imágenes extraídas del mismo video de prueba (pruebas Tipo1), el sistema logra su mayor performance, con una capacidad de reconocimiento aproximadamente de un 73% en los casos que cuenta con el mayor número de imágenes de entrenamiento (NIE=100).
- En las pruebas cuyo entrenamiento fue realizado con imágenes extraídas desde Internet (pruebas Tipo 2), el sistema logra su mayor performance, con una capacidad de reconocimiento aproximadamente de un 68%) en los casos en que cuenta con un entrenamiento de solo las 20 o 30 imágenes cualitativamente mas cercanas al video de prueba.

Dado lo anterior, se puede plantear que el sistema desarrollado es altamente sensible a la calidad de la información utilizada en el entrenamiento. Dicha calidad se entiende por el grado de similitud en luminosidad, pose y gesto respecto al video de prueba.

Líneas de Trabajo Futuras

El trabajo futuro se visualiza focalizado en los siguientes puntos:

- Utilización de metodologías alternativas para el reconocimiento facial.

Mediante esta línea de desarrollo se pueden buscar mejoras en los resultados del sistema desarrollado, utilizando algoritmos que logren una mayor performance de reconocimiento tanto en imágenes de rostro frontal como en imágenes de con mayores grados de inclinación, rotación o giro que los admitidos por la metodología PCA.

- Aplicar funciones adicionales para normalización de imágenes de rostro.

En este trabajo se realiza un pre-procesamiento de las imágenes de rostro mediante la normalización en tamaño y color, características que podrían ser complementadas con funciones que corrijan o estandaricen por ejemplo el ángulo de giro de un rostro. Con este solo elemento adicional, eventualmente se amplía considerablemente el conjunto de imágenes o cuadros del video sobre los cuales se realiza la detección de rostro.

- Estudiar el comportamiento del sistema desarrollado en función de la pose del rostro.

La herramienta para generación de referencia *Ground Truth* desarrollada en este trabajo, permite caracterizar la aparición del rostro objetivo en el video, en base la pose del rostro, es decir, el grado de inclinación, rotación o giro (ver punto 7 Anexo 4). Esta información permitiría el desarrollo de un estudio que determine las capacidades de acierto del sistema para las distintas posibilidades de posición del rostro, lo cual podría ser utilizado para determinar el comportamiento o ámbitos de mejora que aportan algoritmos de reconocimiento facial alternativos.

Referencias

- [1] Bargueño MJ, García-Bastos JL, González-Buitrago JM. Las curvas ROC en la evaluación de las pruebas diagnósticas, *Medicina Clínica*, 1995; 104, 17.
- [2] Roger Gimeno, Estudio De Técnicas De Reconocimiento Facial. Tesis (Magíster en Ciencias). Barcelona, España. Universidad de Cataluña, Departamento de Señales y Comunicaciones, 2010. 86 h.
- [3] Gary Bradski, Adrian Kaehler. *Learning OpenCV: Computer Vision With The OpenCV Library*. O'reilly Media, Octubre, 2008.
- [4] L. Torres, L. Lorente, Josep Vila, “Automatic Face Recognition Of Video Sequences Using Selfeigenfaces”, Department Of Signal Theory And Communications polytechnic University Of Catalonia Barcelona, España, Diciembre 2000.
- [5] Julian Smart, Kevin Hock. *Cross-Platform GUI Programming with wxWidgets*. Prentice Hall Professional Technical Reference, USA, Julio, 2005.
- [6] M. A. Turk, A. P. Pentland. Face recognition using eigenfaces. *Proceedings of the IEEE Computer Society Conf. on Computer Vision and Patter Recognition*. pp. 586-591, Maui, Hawaii, 1991.
- [7] Mislav Grgic, Kresimir Delac, Face Recognition Home Page [en línea] <[Http://Www.Face-Rec.Org](http://www.face-rec.org)> [consulta: julio 2012]
- [8] Thomas Heseltine. Evaluation of Image Pre-processing Techniques for Eigenface Based Face Recognition. The University of York, Department of Computer Science, 2005.
- [9] Jorge García Bueno. Enhancing face recognition by image warping. Master Thesis in Electronics and Electrical Engineering. University of Glasgow, Scotland, Department of Electronics and Electrical Engineering, 2009.
- [10] Paul Viola, Michael J. Jones. Rapid Object Detection using a Boosted Cascade of Simple features. *Conference On Computer Vision And Pattern Recognition*, 2001.
- [11] Amol Ambardekar, Mircea Nicolescu, And Sergiu Dascalu. Ground Truth Verification Tool (Gtv) For Video Surveillance Systems. Department Of Computer Science And Engineering University Of Nevada, Reno, USA, 2009.
- [12] Xiaozheng Zhang, Yongsheng Gao. Face recognition across pose: A review. *Pattern Recognition Volume 42*, 2009.
- [13] Sanhueza Oscar. Análisis y Detección de Características Faciales Usando Aprendizaje Estadístico Universidad de Chile Facultad de Ciencias Físicas Y Matemáticas Departamento De Ingeniería Eléctrica, Santiago, Chile, Marzo 2008.

- [14] Ashwin Venkatraman. Face Recognition in Videos. Imperial College. University Of London, June 14, 2006
- [15] UMIST Face Database. <http://images.ee.umist.ac.uk/danny/database.html>.
- [16] M. Turk, A. Pentland. Face Recognition Using Principal Components Analysis. Eigenfaces for Recognition, Journal of Cognitive Neuroscience, 3(1), pp. 71-86, 1991.
- [17] Carmen Au, Jean-Sebastien Legare, Reehan Shaikh. Face Recognition: Robustness of the Eigenface Approach. McGill University, Montréal, School of Computer Science & Center for Intelligent Machines. Québec, Canada, 2009.
- [18] L. Lorente and L. Torres. A Global Eigen Approach for Face Recognition. In International Workshop on Very Low Bit-rate Video Coding. Urbana, Illinois, USA, 1998.
- [19] Tom Fawcett. ROC Graphs: Notes and Practical Considerations for Researchers. http://home.comcast.net/~tom.fawcett/public_html/papers/ROC101.pdf. MS 1143, 1501 Page Mill Road, Palo Alto, CA 94304. March 16, 2004
- [20] Origins of Automated Facial Recognition (1964 – 1966) [En línea] <<http://www.historyofinformation.com/expanded.php?id=2495>>. [consulta: agosto 2012].
- [21] Biometría Tu Eres la Llave (2010) [En línea] <<http://www.gigabytes.cl/biometria-tu-eres-la-llave.php>>. Revista digital de tecnología. [consulta: agosto 2012]
- [22] Google Incorpora Tecnología de Reconocimiento facial (2011) [En línea] <<http://www.libertaddigital.com/internet/2011-12-09/google-incorpora-la-escalofriante-tecnologia-de-reconocimiento-facial-en-las-fotos-1276443802/>> [consulta: Septiembre 2011]
- [23] Facebook comprará firma de reconocimiento facial en fotos (2012) [En línea] <<http://noticiascl.terra.cl/tecnologia/noticias/0,,OI5844492-EI12470,00-Facebook+comprara+firma+de+reconocimiento+facial+en+fotos.html>> [consulta: julio 2012]
- [24] Últimos Avances en Biometría (2011) [En línea] <http://papeldigital.info/eespecialesal/index.html?2011082201#> [consulta: enero 2012].
- [25] Luis Torres. Is There Any Hope For Face Recognition. [En línea] <<http://www.face-rec.org/interesting-papers/General/cr1182.pdf>>. Technical University of Catalonia, Barcelona, Spain, 2008 [consulta: agosto 2012].
- [26] Teresa Noguera. Curvas ROC e interpretación de AUC. [En línea] <http://eio.usc.es/pub/mte/descargas/ProyectosFinMaster/Proyecto_418.pdf> Facultad de Matemáticas. Universidad de Santiago de Compostela. 2010 [consulta: Enero 2012].
- [27] D. Carrero, B. Ruíz, L. Puente y M.J. Poza. Prestaciones de la Normalización del Rostro en el Reconocimiento Facial. [En línea]. <[61](http://e-</p>
</div>
<div data-bbox=)

archivo.uc3m.es/bitstream/10016/9489/3/prestaciones_carrero_JRBP2010.pdf>.
Universidad Carlos III de Madrid. Madrid. 2009 [consulta: Enero 2012].

- [28] Reconocimiento Facial y Marketing [En línea] <<http://dis.um.es/~ginesgm/files/inv/pfc-baru/pfc-baru.pdf>> [consulta: diciembre 2011]
- [29] Reconocimiento facial en Seguridad [En línea] <<http://www.csospain.es/EI-FBI-invierte-1.000-millones-en-un-sistema-de-reconocimien/seccion-actualidad/noticia-125601>> [consulta: diciembre 2011]

ANEXO A: Especificaciones Ambiente de Desarrollo y Pruebas.

Introducción

A continuación se detalla el ambiente de hardware y software utilizado para el desarrollo y pruebas del sistema de reconocimiento facial desarrollado. Se entrega además información respecto a fuentes de información y sitios de Internet en los cuales se puede acceder a las versiones de software utilizadas para el caso del IDE de desarrollo, Compilador y Librerías OpenCV.

a) Hardware Utilizado

Computador: Marca HP Compac, Modelo DC7900
Memoria: 3,5 GB RAM
Procesador: Intel(R) Core(TM) Quad Q9400 2,66 GHz

b) Sistema Operativo

Windows XP Profesional Versión 2002 Service Pack 3

c) Plataforma de Desarrollo C++

IDE: WxDev-C++, Versión 7.3.1.3
Website: <http://wxdsgn.sourceforge.net/>
Book online: <http://wxdevcpp-book.sourceforge.net/>
Download: <http://wxdsgn.sourceforge.net/?q=node/4>

d) Compilación C++

Compilador: MINGW (Minimalist GNU for Windows)
Website: <http://www.mingw.org/>
Book online: <http://www.mingw.org/wiki>

Download: <http://sourceforge.net/projects/mingw/files/MinGW/>

e) Librerías Procesamiento de Imágenes OpenCV

Librería: OpenCV, Versión 2.1.0 pre-compilada para Windows XP

Website: <http://www.opencv.org/>

Book online: <http://opencv.willowgarage.com/wiki/>

Download: <http://sourceforge.net/projects/opencvlibrary/files/opencv-win/2.1/>

f) Software para construcción de Gráficos

Se utilizó la herramienta Graph 4.1, el cual es un S/W de acceso libre (www.padowan.dk).

ANEXO B: Funciones OpenCV utilizadas para el Procesamiento de Imágenes

Introducción

A Continuación se detallan las funciones pertenecientes al paquete de librerías OPENCV versión 2.2.0, que fueron utilizadas para el manejo y procesamiento de imágenes del sistema de reconocimiento facial desarrollado.

1. Creación de imágenes

CvCreateImage: función que crea una estructura lógica en memoria capaz de cargar el contenido y metadatos de una imagen digital, con el fin facilitar su procesamiento.

2. Copia de imágenes

- a) CvCopy: función que copia una imagen ya cargada en memoria a partir de la función CvCreateImage hacia otra estructura creada también con la función CvCreateImage.
- b) CvCloneImage: Similar a la función anterior, pero utiliza un mecanismo de copia directo, por lo cual se asimila a un clonado de imagen. Tiene mejor performance.

3. Detección de objetos (rostro) en una imagen

CvHaarDetectObjects función que contiene implícito al algoritmo de detección de rostros dentro de una imagen. Recibe como argumentos la imagen sobre la cual se requiere detectar, la referencia al entrenamiento del tipo de objetos que se desea

4. Lectura de un archivo de video

- a) CvCaptureFromAVI: Función que carga en una estructura lógica de memoria un archivo de video en formato .avi
- b) CvRetrieveCuadro : Función que retorna secuencialmente las imágenes correspondientes a los cuadros de un archivo de video.
- c) CvQueryCuadro: función similar a CvRetrieveCuadro pero con la cual se obtiene una mejor performance.

5. Creación de una zona rectangular sobre una imagen

CvRectangle: función que marca un rectángulo en una zona definida de una imagen. Recibe como argumento la imagen, el tamaño del rectángulo y su ubicación.

6. Ecualización de histograma

CvEqualizeHist: Función que realiza la ecualización del histograma de una imagen de entrada.

7. Guardar/cargar imagen en/desde un archivo

- a) CvSaveImage: función utilizada salvar una imagen a un archivo físico en disco.
- b) CvLoadImage: función utilizada para cargar en memoria una imagen contenida en un archivo físico en disco.

8. Liberar memoria utilizada en imagen

CvReleaseImage: Función que limpia la estructura lógica en memoria definida para contener una imagen mediante la función CvCreateImage.

9. Cambia escala de color de una imagen

CvCvtColor: configura la escala cromática que se desea utilizar para el despliegue de una imagen. En particular se ha utilizado para llevar imágenes en colores a niveles de gris.

10. Deplegar una imagen en una ventana

- a) CvNamedWindow: Crea una ventana en la cual se podrá desplegar una imagen. Recibe como argumentos el tamaño y la ubicación deseada.
- b) CvShowImage: Despliega una imagen dentro de una ventana creada con la función CvNamedWindow.
- c) CvDestroyWindow: Borra una ventana creada con CvNamedWindow.

ANEXO C: Algoritmo de Detección Facial

Introducción

A continuación se detalla a nivel teórico el algoritmo de detección facial utilizado en el desarrollo de esta tesis.

Algoritmo de Detección Facial basado el filtros Haar y clasificadores en cascada.

El algoritmo utilizado para la detección de rostros, tanto en la etapa de entrenamiento como en la etapa de reconocimiento en video, se basa en la extracción de características con filtros de ‘base Haar’ y clasificadores en modo cascada. Dicho método, llamado ‘Clasificadores Haar’, fue propuesto por Paul Viola y Michael Jones [10] el cual es un clasificador basado en árboles de decisión con entrenamiento supervisado.

Básicamente, mediante la detección de características locales basadas en el cambio de intensidad se puede obtener un detector de caras robusto, técnica que hoy en día es utilizada incluso no solo para detectar rostros sino también cuerpos rígidos.

Este clasificador funciona a partir de un entrenamiento previo realizado con unos cientos de vistas de ejemplos de un objeto en particular (ej. una cara o un automóvil), llamados ejemplos positivos, y también entrenado con ejemplos negativos. Luego que el clasificador es entrenado, puede ser aplicado a regiones de interés en una imagen de entrada. La salida del clasificador marca “1” si la región es congruente con el objeto (ej. cara/automóvil), y “0” en el caso contrario. Para buscar el objeto en la totalidad de la imagen, se mueve la ventana de búsqueda a lo largo de la imagen y así revisa cada sector.

El clasificador ha sido diseñado para que pueda ser fácilmente “redimensionado” en orden de ser capaz de encontrar objetos de interés a diferentes tamaños, lo cual es más eficiente que redimensionar la imagen por sí misma. Por lo tanto, para encontrar un objeto de tamaño desconocido en la imagen, la búsqueda se realiza varias veces a diferentes escalas.

La denominación “cascade” o cascada en el nombre del clasificador significa que el clasificador resultante consiste en varios clasificadores simples que son aplicados subsecuentemente a una región de interés hasta que en alguna etapa el candidato es rechazado o todas las etapas son aceptadas.

Los clasificadores simples son árboles de decisión con al menos 2 ramas. Las Características Haar son la entrada de dichos clasificadores.

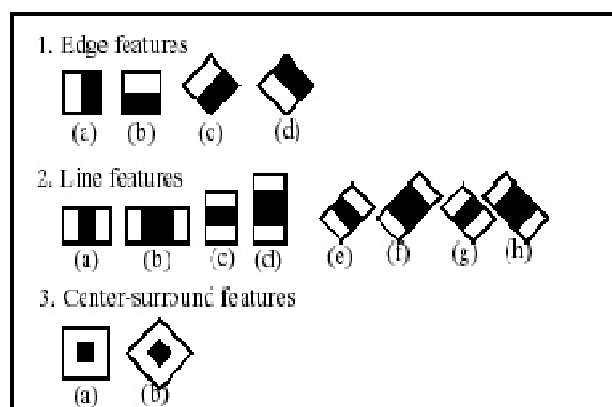


Figura 31: Características Haar

La Figura 1 muestra un ejemplo de características Haar. Las usadas en un clasificador en particular dependen de su forma (1a, 2b, etc.) posición en la región de interés y la escala (no es la misma escala usada en la etapa de detección). Por ejemplo, en el caso de la característica (2c) la respuesta es calculada como la diferencia entre la suma de los píxeles de la imagen bajo el rectángulo cubriendo la característica completa (incluyendo las franjas blancas y negras) y la suma de los píxeles de la imagen bajo la franja negra multiplicada por 3 en orden de compensar la diferencia de tamaños entre áreas. La suma de los valores de los píxeles sobre las regiones rectangulares es calculada rápidamente usando imágenes integrales.

Metodología Haar Cascade

La metodología se divide en tres etapas como se muestra en la figura 32: en la primera se realiza una transformación de la imagen generando una nueva llamada imagen integral; en la segunda etapa se realiza la extracción de características usando filtros con base Haar; y por último se usa boosting para la construcción de clasificadores en cascada.

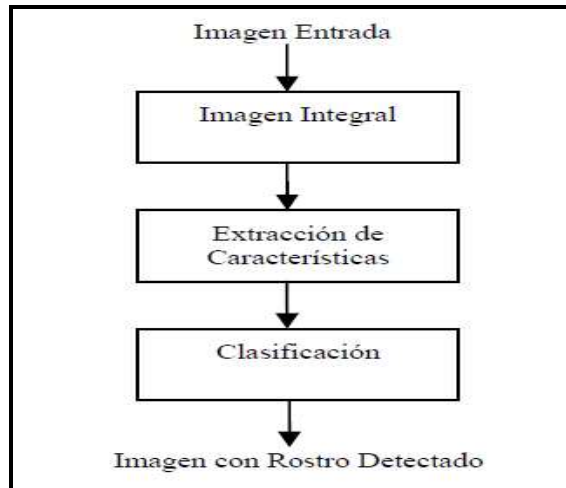


Figura 32: Etapas metodología Haar.

Imagen Integral

Esta imagen permite extraer de forma rápida características a diferentes escalas ya que no se trabaja directamente con los valores de intensidad si no con una imagen acumulativa que se construye a partir de operaciones básicas.



Figura 33: Imagen Integral.

La imagen integral (figura 33), en la localización x, y , contiene la suma de los píxeles de la parte superior izquierda de la imagen y se puede calcular como se indica a continuación:

$$II(x, y) = \sum_{x' \leq x, y' \leq y} Im(x', y')$$

Donde $I(x,y)$ es la imagen integral e $Im(x,y)$ es la imagen original.

a3. Extracción de Características

Las características de cada objeto se extraen al aplicar ciertas funciones que permitan la representación y descripción de los objetos de interés de la imagen (patrones). La extracción de características es un paso en el reconocimiento de patrones en el cuál las medidas u observaciones son procesadas para encontrar atributos que puedan ser usados para asignar los objetos a determinada clase. En la metodología seguida, la extracción de características es realizada aplicando a la imagen filtros con bases Haar. Estos filtros pueden ser calculados eficientemente sobre la imagen integral, son selectivos en la orientación espacial y frecuencia, y permiten ser modificados en escala y orientación. En la figura 3, se muestran algunos de los filtros usados para la extracción de características. Los filtros con bases Haar, realizan una codificación de diferencia de intensidades en la imagen, generando características de contornos, puntos y líneas, mediante la captura de contraste entre regiones.

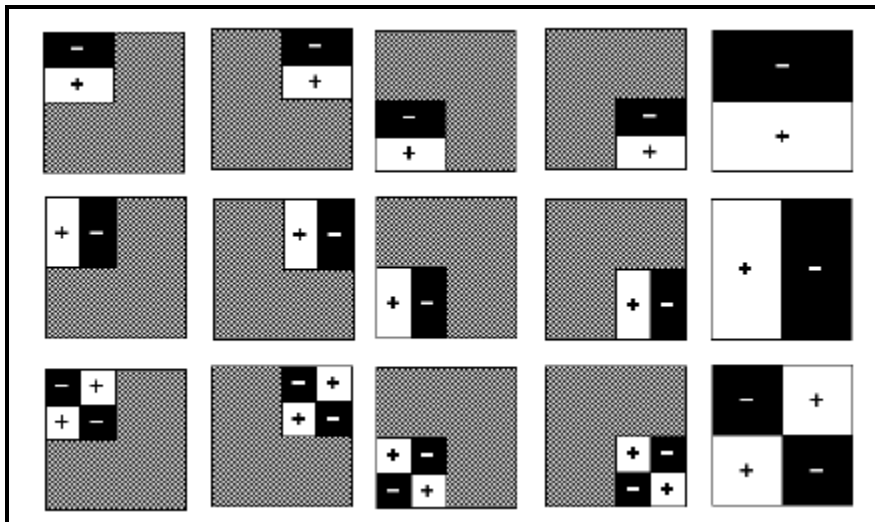


Figura 34: Filtros Haar rotados, trasladados y con cambios de Escala.

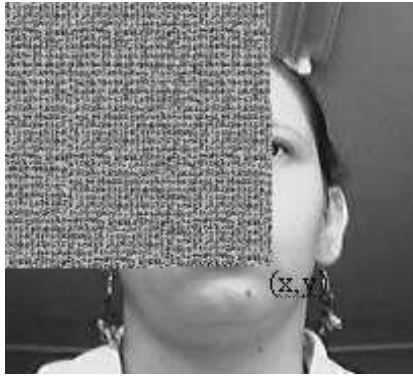


Figura 35: Convolución del filtro Haar con una imagen integral.

En la figura 35, se muestra la convolución de un filtro Haar con la imagen integral. De esta operación se puede extraer una característica en un tiempo constante sobre la imagen integral adicionando y sustrayendo los valores de los vértices para cada rectángulo. Para mayor claridad, en la figura la suma de los píxeles que forman el rectángulo D se puede calcular como:

$$sumD = (4 + 1) - (2 + 3)$$

Donde 1, 2, 3, 4, son los valores dados en la imagen integral en dichas localizaciones.

a4. Clasificación

Esta etapa dentro del algoritmo de detección se encarga de asignar un conjunto de características dado a una clase con la que se encuentra una mayor similitud, de acuerdo a un modelo inducido durante el entrenamiento. Boosting es un método de clasificación que combina varios clasificadores básicos para formar un único clasificador más complejo y preciso. La idea se basa en la afirmación de que varios clasificadores sencillos, cada uno de ellos con una precisión ligeramente superior a una clasificación aleatoria, pueden combinarse para formar un clasificador de mayor precisión, siempre y cuando se disponga de un número suficiente de muestras de entrenamiento.

En la figura 36, se muestra un esquema de un clasificador en cascada.



Figura 36: Clasificador en cascada.

Para aplicar la técnica de *boosting* primero se debe establecer un algoritmo de aprendizaje sencillo (clasificador débil o base), que será llamado repetidas veces para crear diversos clasificadores base. Para el entrenamiento de los clasificadores base se emplea, en cada iteración, un subconjunto diferente de muestras de entrenamiento y una distribución de pesos diferente sobre las muestras de entrenamiento. Finalmente, estos clasificadores base se combinan en un único clasificador que se espera sea mucho más preciso que cualquiera de los clasificadores base por separado. En función de los clasificadores base que se utilicen, las distribuciones que se empleen para entrenarlos y el modo de combinarlos, podrán crearse distintas clases del algoritmo genérico de *boosting*. El algoritmo de *boosting* empleado por Viola y Jones en su trabajo es conocido como *AdaBoost*.

ANEXO D: Manual de Usuario del Sistema

Introducción

El programa se divide en 8 módulos: Entrenamiento, Video, Parámetros, procesamiento, Resultados, Ayuda, Herramienta Ground Truth. A continuación se presenta una descripción detallada de las funcionalidades de cada módulo así como los archivos y carpetas de importancia que deben existir para el correcto funcionamiento del sistema.

1. Inicio

En la figura 37 presenta la interfaz de inicio del sistema, en el cual se muestra un diagrama funcional con las descripciones de cada módulo:

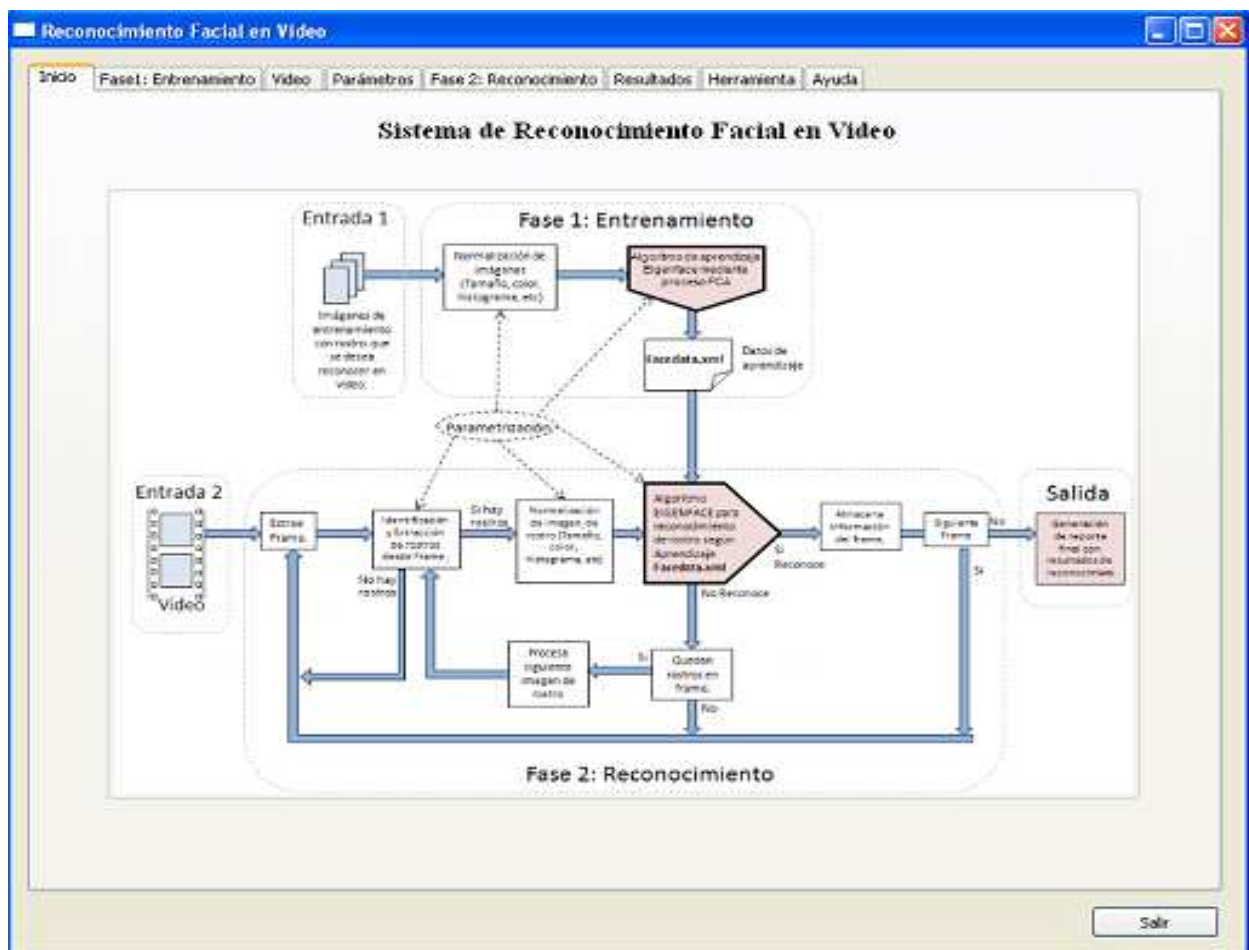


Figura 37: Módulo de inicio

2. Módulo de Entrenamiento

Encargado de generar la base de datos con la información de entrenamiento, a partir de un conjunto de imágenes del rostro que se desea identificar en el video. Contiene las siguientes funciones accionadas por 5 botones: Cargar Imagen, Eliminar, Borrar Todos, Cargar Lista, Entrenar.

Su interfaz corresponde a la presentada en la figura 38.

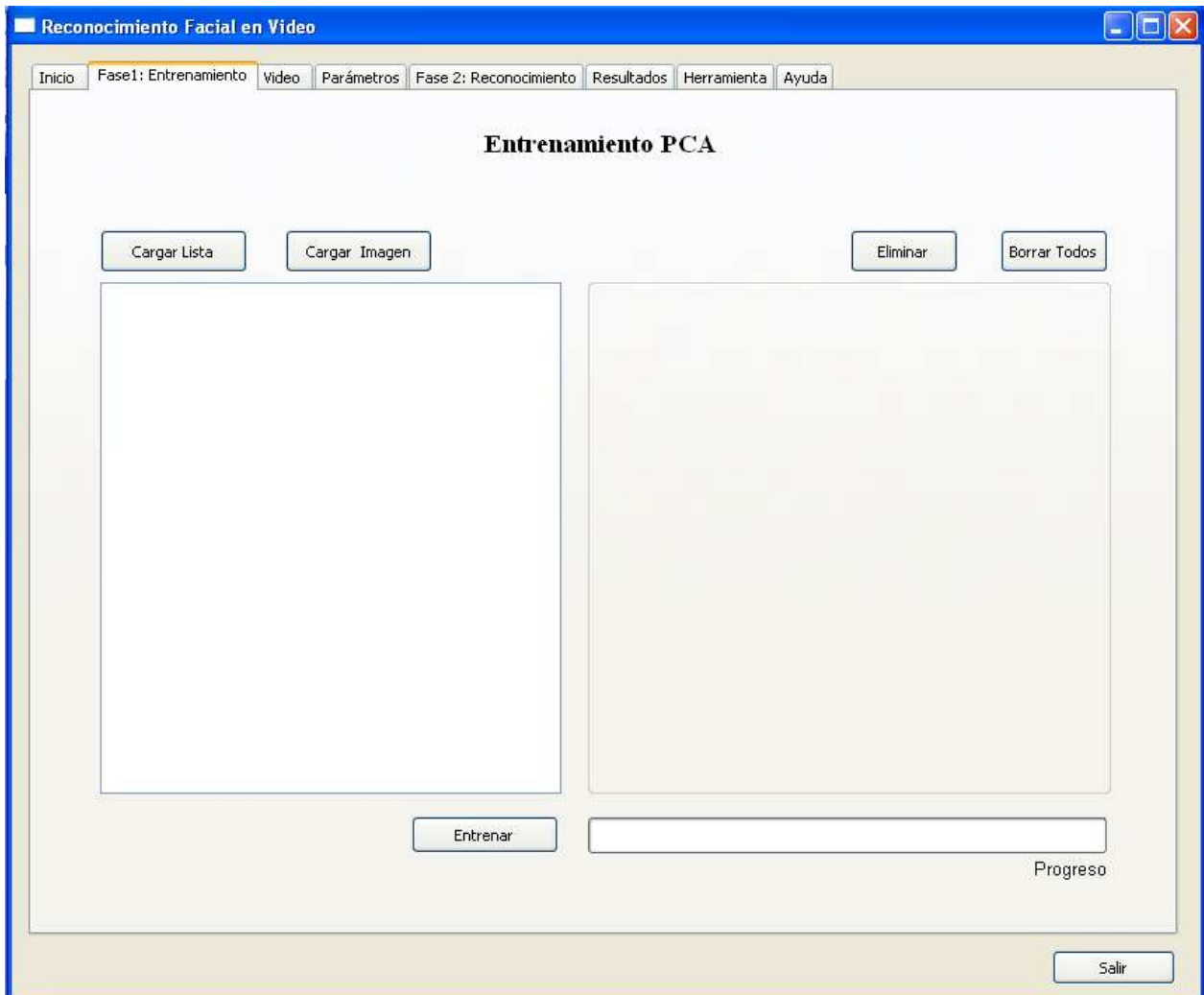


Figura 38: Módulo de entrenamiento.

Cargar Imagen

Permite cargar un archivo de imagen para ser agregado a la lista de entrenamiento.

Eliminar

Permite la eliminación de un archivo de imagen previamente seleccionado con el mouse.

Borrar Todos

Permite la eliminación de todos los archivos de imagen previamente cargados.

Cargar Lista

Permite seleccionar un archivo de texto que contenga el listado de archivos de imágenes que se deseen utilizar como entrenamiento. Este archivo debe contener una línea por cada archivo de imagen y debe contener la ruta completa de acceso. Ejemplo:

C:/Documentos/Prueba/imagen1.bmp

Entrenar

Inicia el proceso de entrenamiento a partir del total de los archivos cargados. Luego de realizado el entrenamiento, genera un archivo en formato XML llamado facedata.xml. Este archivo es generado en el mismo directorio de funcionamiento del sistema.

Barra de Estado

Barra que indica el nivel de avance del proceso entrenamiento y generación de la base de datos de entrenamiento facedata.xml. Cuando se completa entera de color verde, termina el proceso desplegando un mensaje acerca del estado de la finalización del proceso.

3. Módulo de Video

Encargado del ingreso del archivo de video que se desea procesar. Entrega otras facilidades como la de definir sectores o escenas del video que se desee analizadas en forma individual. Contiene las siguientes funciones accionadas por 6 botones: Cargar Video, Selección de Escenas, Inicio, Fin, Eliminar, Eliminar Todos.

Su interfaz corresponde a la presentada en la figura 39.

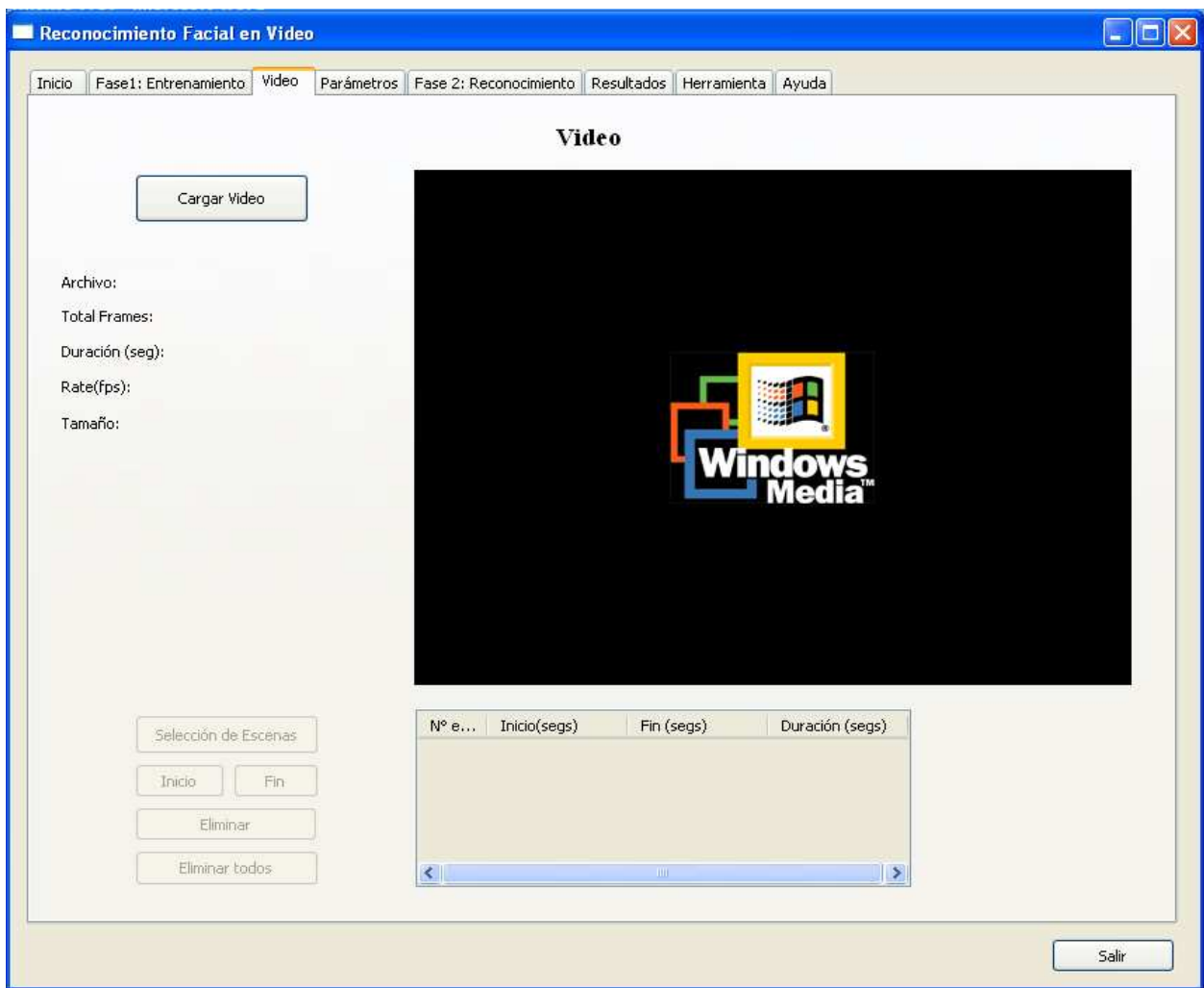


Figura 39: Módulo de video.

Cargar Video

Permite cargar el archivo de video que se desea analizar para reconocimiento de rostro. Una vez que este ha sido cargado, en la parte izquierda superior de la ventana se despliega los siguientes metadatos del video:

- *Archivo*: Ruta y Nombre del archivo cargado.
- *Total cuadros*: Número total de cuadros que contiene el video.
- *Duración (segundos)*: Duración en segundos del video.
- *Rate (fps)*: Tasa de reproducción en Cuadros por segundo.

Selección de Escena

Habilita la posibilidad de marcar escenas dentro del video con el fin de restringir la búsqueda del rostro solo a estas. En el panel de escenas se despliega información referente a cada escena definida: N° Orden, Segundos de Inicio, Segundos de Fin, Duración de la escena.

Inicio:

Permita marcar el tiempo de inicio de una escena.

Fin:

Marca el tiempo de Fin de una escena.

Eliminar:

Permite la eliminación de una escena previamente seleccionada con el mouse.

Eliminar Todos:

Permite la eliminación de todas las escenas definidas previamente.

4. Módulo de Configuración

Encargado de proveer una interfaz de configuración de los parámetros relevantes relacionados con el funcionamiento del sistema. Su interfaz corresponde a la presentada en la figura 40.

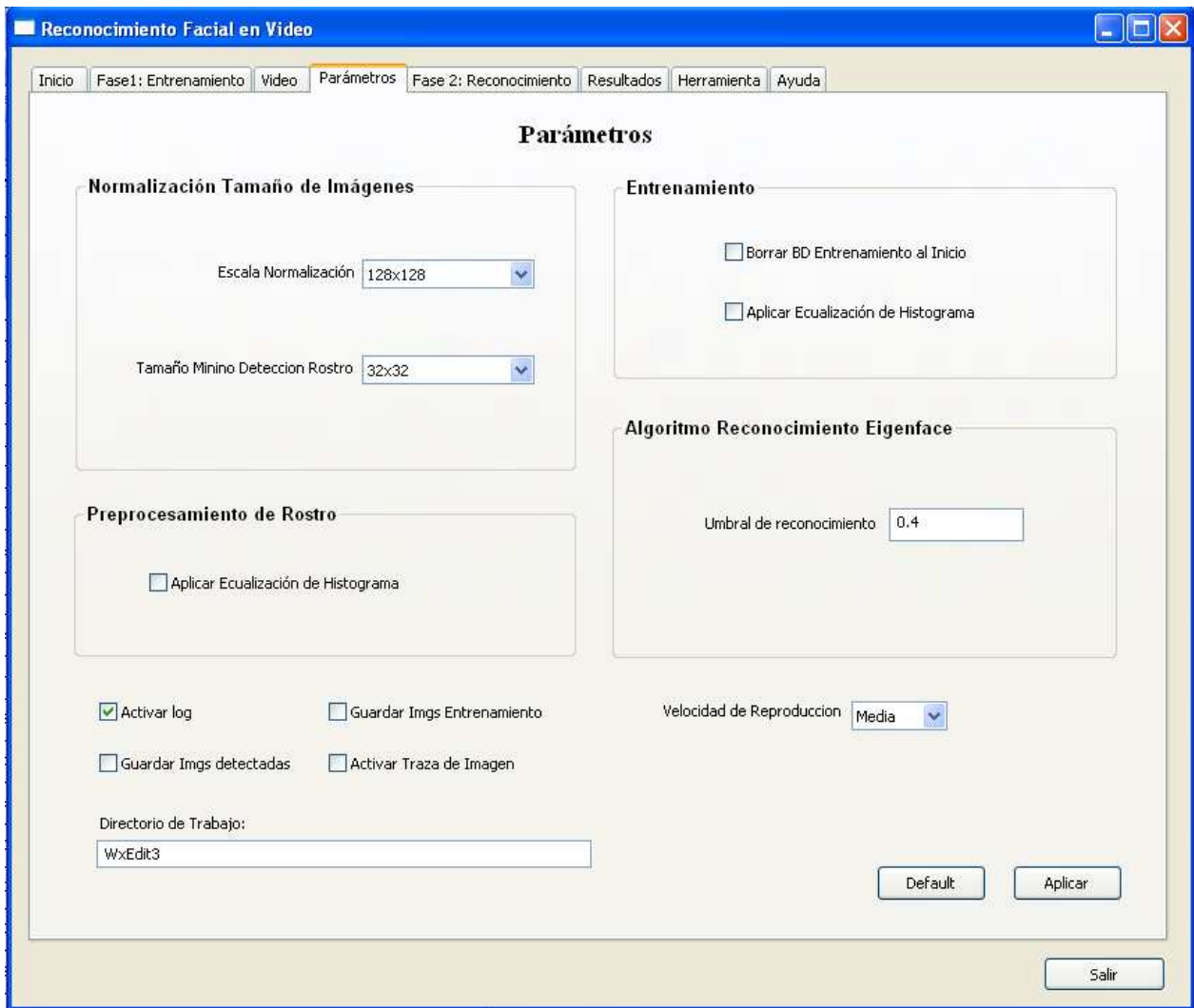


Figura 40: Módulo de Configuración

Normalización tamaño de Imágenes

- **Escala normalización:** Caja de opciones en la cual se puede seleccionar la escala de normalización que será aplicado a las imágenes de rostro de entrenamiento y a las imágenes de rostro detectadas en los cuadros de video. Las opciones disponibles son 64x64, 128x128, 256x256. El valor por defecto es 128x128.
- **Tamaño mínimo detección rostro:** Caja de opciones con la cual se puede seleccionar el tamaño mínimo de rostro a ser detectado tanto en las imágenes de entrenamiento como en los cuadros de video analizado. Las opciones disponibles son 64x64, 128x128, 256x256. El valor por defecto es 128x128.

Entrenamiento

- **Borrar base de datos al inicio:** flag con el cual se indica si se borra o no el archivo facedata.xml al iniciar el programa. Al activarlo, el archivo se borrará y se debe iniciar realizando un entrenamiento nuevo. El valor por defecto es desactivado.
- **Aplicar ecualización de histograma:** flag con el cual se indica si se desea o no activar la ecualización de histograma sobre las imágenes ingresadas para realizar el entrenamiento. El valor por defecto es desactivado.

Pre-procesamiento de Rostro

- **Aplicar ecualización de histograma:** flag con el cual se indica si se desea o no activar la ecualización de histograma sobre las imágenes de rostro detectadas en los cuadros del video analizado. El valor por defecto es desactivado.

Algoritmo Reconocimiento Eigenface:

- **Umbral de reconocimiento (UR):** Caja de opciones con la cual se puede seleccionar el valor de UR para detección de rostro. A partir de este valor, un rostro detectado se considera reconocido respecto el rostro objetivo. Las opciones disponibles son -0.2, -0.1, 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9. El valor por defecto es 0.4.
- **Tipo Distancia:** Caja de opciones con la cual se puede seleccionar el tipo de distancia que se desea utilizar para calcular la cercanía entre la proyección de la imagen analizada y los eigenvectores del entrenamiento. Las opciones disponibles son Euclidiana y Mahalanobis. El valor por defecto es Euclidiana.

Guardar Imágenes Detectadas.

Flag que indica si se desea o no guardar en el directorio \$HOME/rostros/ las imágenes de rostro detectadas los cuadros de video analizados. El valor por defecto es desactivado.

Guardar Imágenes Entrenamiento

Flag que indica si se desea o no guardar en el directorio \$HOME/entr/ las imágenes de rostro detectadas en el proceso de entrenamiento a partir del set de imágenes ingresadas. El valor por defecto es desactivado.

Activar Log

Flag que indica si se desea o no activar la generación del Log de detalles del procesamiento del video analizado. El valor por defecto es activado.

Velocidad de Reproducción

Caja de opciones que permite seleccionar la velocidad de reproducción de video utilizada en las herramientas de visualización existentes en el Módulo de Resultados. Las opciones disponibles son Baja, Media, Alta. El Valor por defecto es Media.

Default

Botón con el cual el sistema retoma los valores por defecto de cada uno de los parámetros.

Aplicar

Botón que realiza la acción de ‘commit’ de los valores ajustados en el módulo de parámetros. Solo después de realizar esta operación el programa operará con los nuevos valores.

5. Módulo de Procesamiento

Permite dar inicio al proceso de búsqueda del rostro objetivo dentro del video. Para ello dispone de un botón, una barra de estado y 2 indicadores estadísticos. Su interfaz corresponde a la presentada en la figura 41.

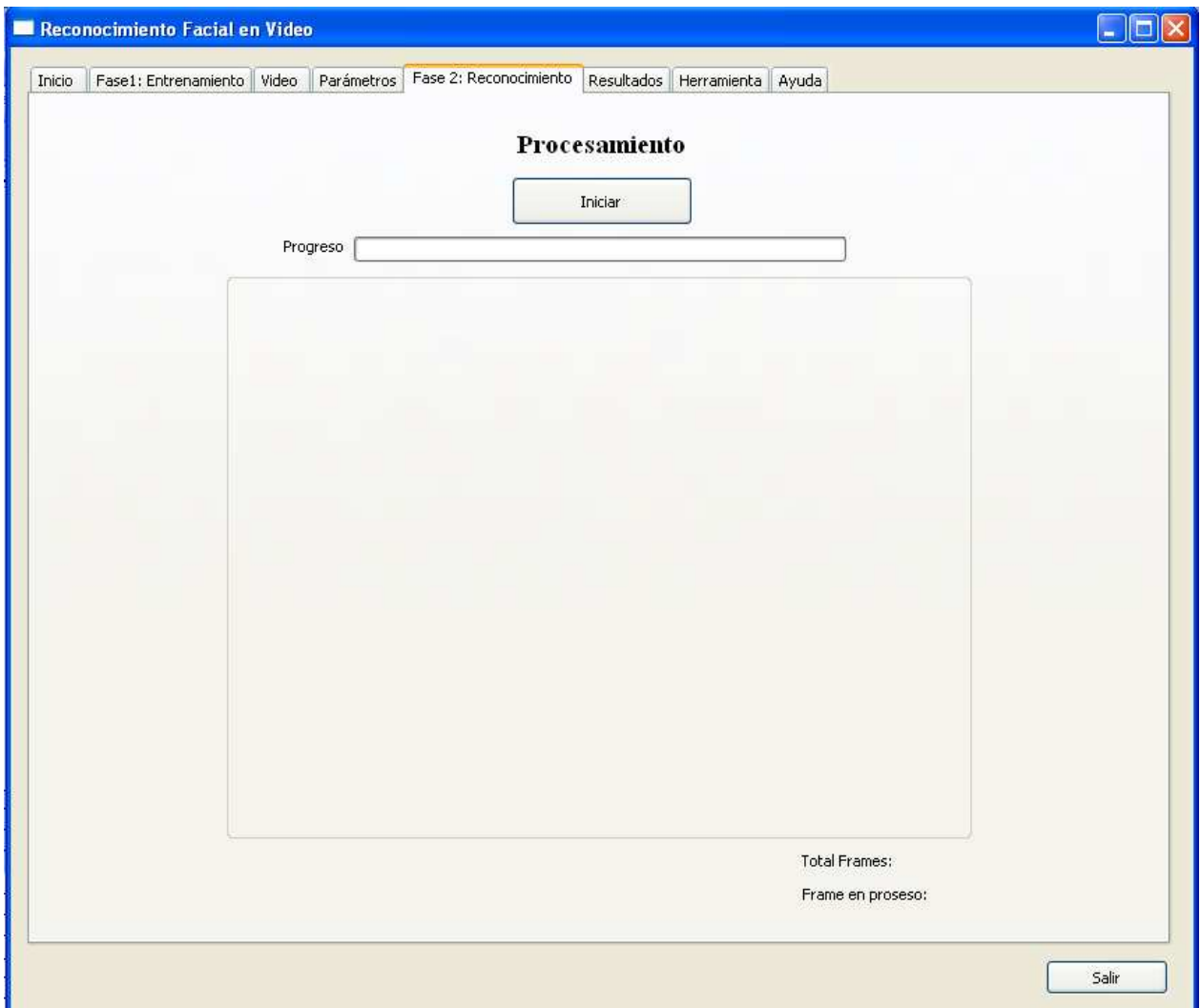


Figura 41: Módulo de Procesamiento.

Iniciar Procesamiento: Botón que da inicio al proceso de análisis de cuadros del archivo de video, ya sea en las escenas seleccionadas o el video completo.

Barra de Estado: Indicador visual del porcentaje de avance en el procesamiento del archivo de video analizado.

Cuadro en Proceso: Indica numero de orden el cuadro que se halla actualmente en proceso de detección de rostro y búsqueda de coincidencia con entrenamiento.

Total Cuadros: Despliega el número total de cuadros que contiene el archivo de video analizado.

Se despliega un cuadro con la imagen del cuadro en proceso, demarcando con un rectángulo rojo el o los rostros detectados.

6. Módulo de Resultados

Permite la visualización, a nivel estadístico y gráfico, del resultado entregado por el sistema luego del procesamiento del video. También permite el despliegue del archivo de log detallado del proceso, además del reporte de aciertos. Su interfaz se presenta en la figura 42.

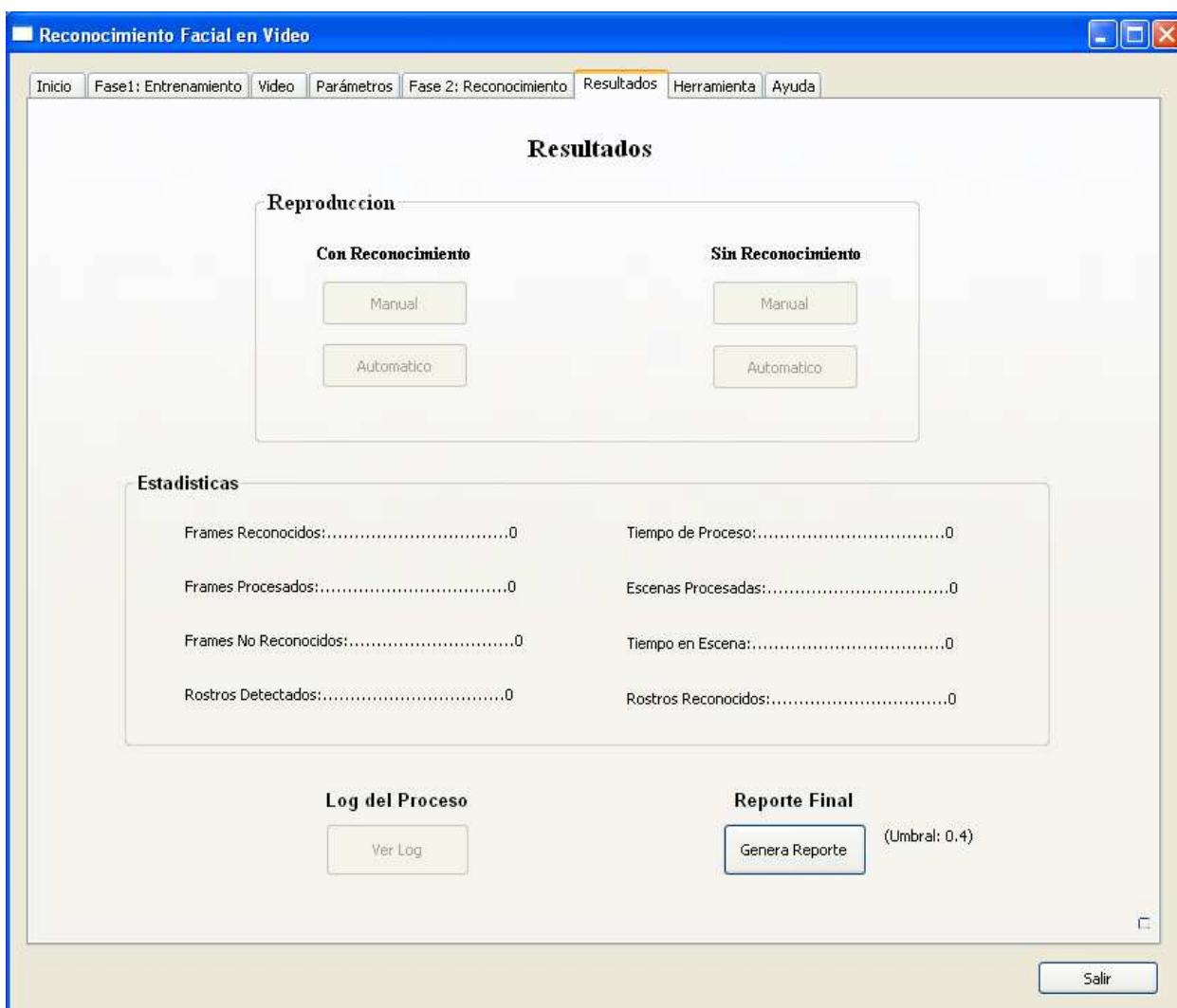


Figura 42: Módulo de Resultados

Este sub-módulo presenta 4 botones con los cuales se pueden visualizar los cuadros del video con y sin reconocimiento de rostro:

Con Reconocimiento

- **Ver cuadros manualmente:** Herramienta que permite, mediante una barra de tracking manual accionada con el puntero de mouse, la visualización de las secciones del video donde SI hubo reconocimiento de rostro. Se consideran solo las escenas seleccionadas previamente para el análisis. Si no hubo selección de escenas se considera el video completo.
- **Solo escenas seleccionadas:** Herramienta que permite, mediante reproducción continua del video, la visualización de las secciones donde si hubo reconocimiento de rostro. Se consideran solo las escenas seleccionadas previamente para el análisis. Si no hubo selección de escenas se considera el video completo.

Sin Reconocimiento

- **Ver cuadros Manualmente:** Herramienta que permite, mediante una barra de tracking manual accionada con el puntero de mouse, la visualización de las secciones del video donde NO hubo reconocimiento de rostro. Se consideran solo la(s) escena(s) seleccionadas previamente para el análisis. Si no hubo selección de escenas se considera el video completo.
- **Solo escenas Seleccionadas:** Herramienta que permite, mediante reproducción continua del video, la visualización de las secciones del video dosnde NO hubo reconocimiento de rostro. Se consideran solo la(s) escena(s) seleccionadas previamente para el análisis. Si no hubo selección de escenas se considera el video completo.

Detalles Estadísticos

- **Cuadros reconocidos:** Contabiliza todos los cuadros en los cuales se identifico el rostro buscado.
- **Cuadros Procesados:** Contabiliza la totalidad de los cuadros analizados.
- **Cuadros No Reconocidos:** Contabiliza todos los cuadros en los cuales no se identifico el rostro buscado.
- **Rostros Detectados:** Contabiliza todos los rostros detectados dentro de todos los cuadros analizados.
- **Tiempo de Proceso:** Tiempo total en segundos que demoro el procesamiento de búsqueda de rostro.
- **Escenas procesadas:** Despliega la cantidad de escenas que fueron analizadas durante el proceso de búsqueda de rostro.

- **Tiempo en escena:** Corresponde al tiempo en segundos que el rostro buscado estuvo en escena a tomando en cuenta la cantidad de cuadros en los cuales fue identificado el rostro.
- **Rostros Reconocidos:** Contabiliza el número de veces que fue reconocido el rostro buscado, dentro de todos los rostros detectados en los cuadros analizados.

Visualización del Log del Proceso

Botón con el cual se despliega automáticamente el Log detallado del procesamiento de video realizado.

Reporte de Aciertos

Botón con el cual se despliega automáticamente el reporte de aciertos del procesamiento de video realizado. Para que se pueda generar el reporte de aciertos, debe existir el archivo GT.txt, el cual contiene la información verdadera sobre las apariciones del rostro objetivo en cada cuadro del video analizado. El archivo GT.txt debe ser generado en el módulo Herramienta Ground Truth.

7. Módulo de Herramienta Ground Truth

Permite la generación de un archivo de referencia llamado GT.txt, el cual contiene la información que el usuario del sistema establece como verdad respecto a la presencia del rostro objetivo en el video. Las funcionalidades de este módulo se activan una vez que el archivo de video se ha cargado en el módulo de video. La interfaz presentada en la figura 43, está compuesta por los siguientes elementos:

- **Ventana para despliegue de cuadro:** Muestra el cuadro que debe ser calificado por el usuario. Se actualiza con el siguiente cuadro cada vez que se asigna una calificación.
- **Botón inicio:** inicia la secuencia de despliegue del video cuadro a cuadro
- **Botón fin:** finaliza la secuencia de despliegue del video.
- **Botones de clasificación:** deben ser pulsados para la clasificación del cuadro desplegado. Las posibilidades corresponden a las presentadas en la tabla 7.

Respecto al rostro objetivo.			Código
No contiene rostros			100
Contiene rostro(s), pero no el objetivo.			200
Contiene rostro objetivo	Imagen Nítida	frontal	400
		Rotado	401
		Rotado leve	402
		Giro	403
		Giro Leve	404
		Inclinado	405
		Inclinado Leve	406
	Imagen con Nitidez Media	frontal	410
		Rotado	411
		Rotado leve	412
		Giro	413
		Giro Leve	414
		Inclinado	415
		Inclinado Leve	416
	Imagen con Baja Nitidez	frontal	420
		Rotado	421
		Rotado leve	422
		Giro	423
		Giro Leve	424
		Inclinado	425
		Inclinado Leve	426

Tabla 7: Posibilidades de clasificación de los cuadros de video de respecto el rostro objetivo.

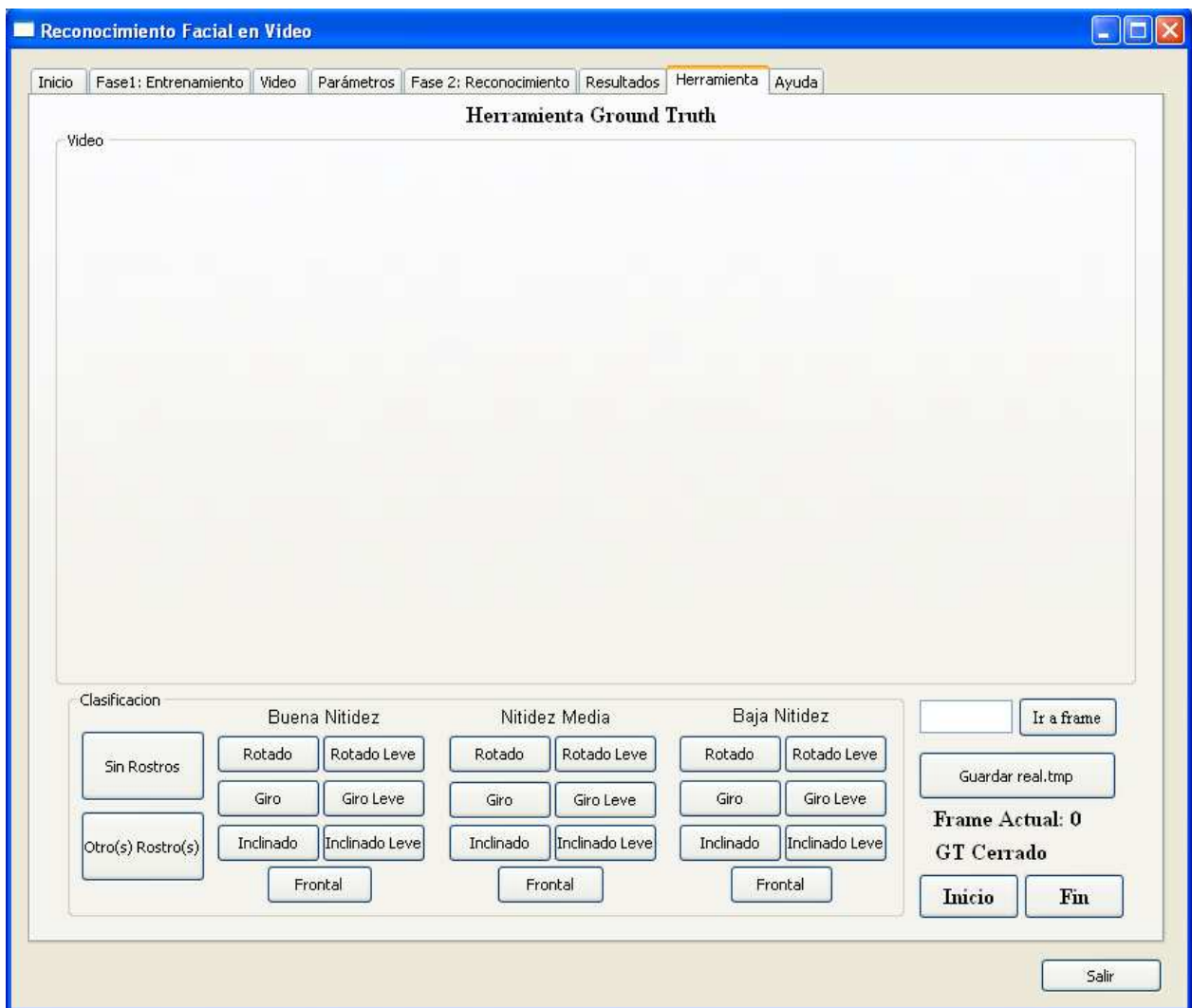


Figura 43: Módulo Herramienta *Ground Truth*

8. Módulo de Ayuda.

Presenta un resumen de información sobre la autoría y versionado del sistema. Además provee un botón mediante el cual el usuario puede desplegar automáticamente el Manual de Usuario del programa. Su interfaz corresponde a la presentada en la figura 44.



Figura 44: Módulo Ayuda

ANEXO E: Formatos de Archivos de Salida

Introducción

Este anexo detalla el contenido de los archivos de salida del sistema de reconocimiento facial, generados a partir del procesamiento del video de prueba y el uso de la herramienta *ground truth*.

a) Archivo proceso.txt

Este archivo es generado a partir del procesamiento de un video de prueba:

- **Encabezado del archivo:**

```
-----  
Reconocimiento de Rostro en Video - RRV V1.0  
Reporte Detallado  
Fecha/Hora: Wed May 30 22:35:00 2012  
Video Procesado : C:\DV\archivosprueba\videos\obama.avi  
# Imágenes de Entrenamiento: 20  
-----
```

- **Descripción campos de datos:**

<i>NºFrm</i>	: Numero secuencial del cuadro en el video
<i>Flg_scn</i>	: Flag que indica
<i>#Rostros_Frm</i>	: Cantidad de rostros detectados en el cuadro
<i>Dist</i>	: Distancia euclideana mínima entre el rostro del cuadro y los eigenvectores
<i>Flg_rec</i>	: Falag que indica reconocimiento: 1=reconocimiento 2= no reconocido
<i>Tpo_Frm</i>	: Tiempo del cuadro dentro del video

- **Fin del archivo:**

```
----- Estadísticas Resultantes -----  
-----  
Cuadros Reconocidos:.....  
Tiempo de Proceso:.....  
Cuadros Procesados:.....  
Escenas Procesadas:.....  
Rostros Detectados:.....  
Rostros Reconocidos:.....  
Cuadros No Reconocidos:.....  
Tiempo en Escena:.....
```

- **Muestra ejemplo del archivo:**


```

Reconocimiento de Rostro en Video - RRV V1.0
Reporte Detallado
Fecha/Hora: Wed May 30 22:35:00 2012
Video Procesado : C:\DV\archivosprueba\videos\obama.avi
# Imagenes Entrenamiento : 20
N°Frm Flg_scn #Rostros_Frm Dist Conf Flg_rec Tpo_Frm
-----

```

```

1 2 0 0 0 0 0.034483
2 2 0 0 0 0 0.068966
3 2 0 0 0 0 0.103448

```

...

```

11935 2 1 5493057.195816 0.528507 1 411.551724
11936 2 1 5564859.428891 0.525436 1 411.586207
11937 2 1 6002835.631606 0.507114 1 411.620690
11938 2 1 5484175.388759 0.528889 1 411.655172

```

...

```

13711 2 0 0 0 0 472.793103
13712 2 0 0 0 0 472.827586
13713 2 0 0 0 0 472.862069

```

```

-----          Estadísticas Resultantes          -----
-----

```

```

Cuadros Reconocidos:.....7036
Tiempo de Proceso:.....8876 segs
Cuadros Procesados:.....13713
Escenas Procesadas:.....0
Rostros Detectados:.....7054
Rostros Reconocidos:.....7036
Cuadros No Reconocidos:.....18
Tiempo en Escena:.....242.62 segs

```

b) Archivo GT.txt

Este archivo es generado a partir del uso de la herramienta Ground Truth:

- **Descripción de campos del archivo:**

- Correlativo Cuadro.
- Flag que indica la presencia (o no) del rostro objetivo en el cuadro.
- Código que caracteriza el rostro observado.

- **Muestra ejemplo del archivo:**

En la figura 45 se realiza una descripción de los campos de datos de cada registro del archivo.

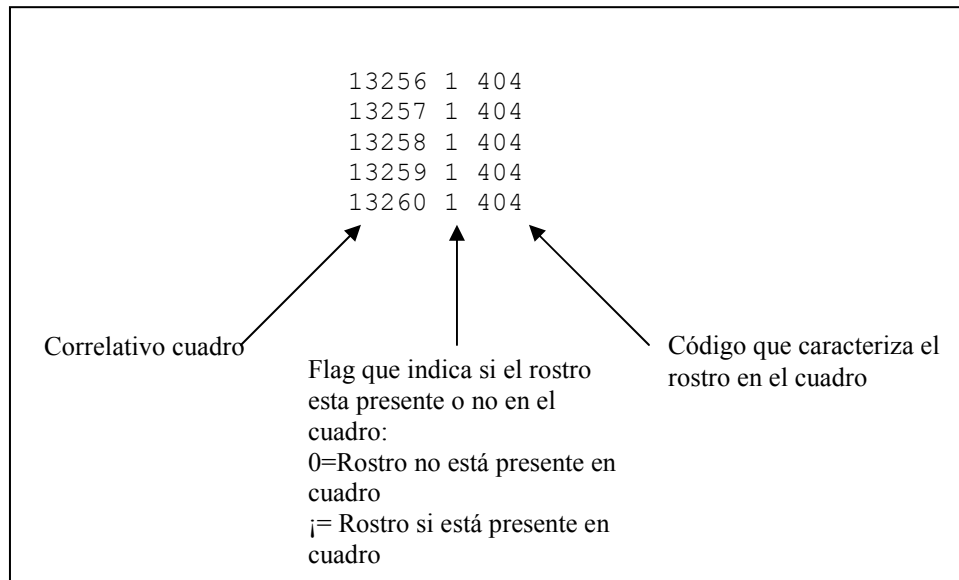


Figura 45: Contenido archivo Ground Truth.

c) Archivo consolidado.txt

Generado a partir del proceso de contrastación entre el archivo proceos.txt y el archivo GT.txt.

- **Descripción de campos del archivo:**

- Valor UR: Umbral del reconocimiento de rostro.
- Verdaderos positivos (VP) : Rostro SI está en cuadro y SI es identificado por el programa
- Verdaderos negativos (VN) : Rostro NO está en cuadro y NO es identificado por el programa
- Falsos positivos (FP): Rostro NO está en cuadro y SI es identificado por el programa
- Falsos negativos (FN): Rostro SI está en cuadro y NO es identificado por el programa
- Uno menos especificidad : $1 - \text{VN} / (\text{FP} + \text{VN})$
- Sensibilidad : $\text{VP} / (\text{VP} + \text{FN})$

- **Muestra ejemplo del archivo:**

En la figura 46 se realiza una descripción de los campos de datos de cada registro del archivo.

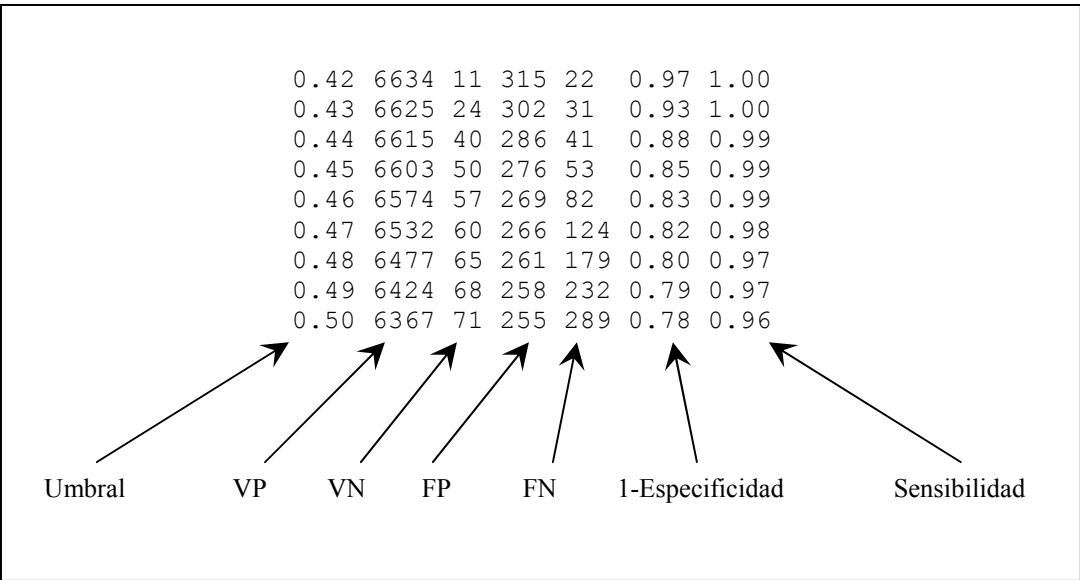


Figura 46: Contenido Archivo Consolidado.txt