



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA
SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN
UNIDAD CULHUACAN

“SECCIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN”

**DETECCIÓN DE IMÁGENES CON CONTENIDO EXPLÍCITO
USANDO LOS MODELOS DE COLOR *HSV* y *YCbCr***

TESIS QUE PARA OBTENER EL GRADO DE:

MAESTRÍA EN INGENIERÍA EN SEGURIDAD Y TECNOLOGÍAS
DE LA INFORMACIÓN

PRESENTA

ING. JORGE ALBERTO MARCIAL BASILIO

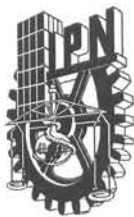
ASESORES:

Dr. GABRIEL SÁNCHEZ PÉREZ
Dr. GUALBERTO AGUILAR TORRES



MÉXICO D.F.

DICIEMBRE DEL 2011



INSTITUTO POLITÉCNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D. F. siendo las 12:00 horas del día 22 del mes de noviembre del 2011 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de SEPI ESIME CULHUACAN para examinar la tesis titulada:

“Detección de Imágenes con Contenido Explícito Usando los Modelos de Color HSV y YCbCr”

Presentada por el alumno:

Marcial	Basilio	Jorge Alberto
Apellido paterno	Apellido materno	Nombre(s)

Con registro:

B	0	9	1	8	0	7
---	---	---	---	---	---	---



aspirante de:

MAESTRÍA EN INGENIERÍA EN SEGURIDAD Y TECNOLOGÍAS DE LA INFORMACIÓN

Después de intercambiar opiniones los miembros de la Comisión manifestaron **APROBAR LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Directores de tesis

 _____ Dr. Gabriel Sánchez Pérez	 _____ Dr. Gualberto Aguilar Torres
---	--

 _____ Dr. Héctor Manuel Pérez Meana	 _____ Dra. Linda Karina Toscano Medina
---	--

 _____ Dr. Mario González Lee	
--	---

PRESIDENTE DEL COLEGIO DE PROFESORES



Dr. Gonzalo Isaác Duchén Sánchez



INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México el día 22 del mes Noviembre del año 2011, el (la) que suscribe Marcial Basilio Jorge Alberto alumno (a) del Programa de Maestría en Ingeniería en Seguridad y Tecnologías de la Información con número de registro B091807, adscrito a la Sección de Estudios de Posgrado de la ESIME unidad Culhuacan, manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección del Dr. Gabriel Sánchez Pérez y del Dr. Gualberto Aguilar Torres y cede los derechos del trabajo intitulado **“Detección de Imágenes con Contenido Explícito usando los modelos de color HSV y YCbCr”**, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección jmarcialb0300@ipn.mx, gasanchezp@ipn.mx y gaguilar@ipn.mx. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.


Jorge Alberto Marcial Basilio

Nombre y firma

A mis padres y a mi familia

AGRADECIMIENTOS

RESUMEN

En el presente trabajo se presenta una investigación para la detección de imágenes con contenido explícito en dispositivos de almacenamiento masivo.

Este trabajo está basado principalmente en el método de detección de piel, para ello se requiere de la utilización de distintas técnicas, como reconocimiento de patrones en el modelo de color HSV o YCbCr.

Además se propone un umbral de clasificación de píxeles para tonos de color piel para los dos modelos de color utilizados en este trabajo, esto para generar un clasificador, que permite discriminar todos los objetos dentro de la imagen analizada.

El sistema propuesto se comparó con dos sistemas comerciales, los cuales cumplen el mismo objetivo que el presentado en esta tesis. Finalmente se pudo comprobar que el sistema propuesto logra detectar las imágenes bajo estudio de mejor forma que los dos sistemas comerciales más usados por investigadores forenses, por lo que el método propuesto puede aplicarse para análisis forense informático o en detección de imágenes pornográficas almacenadas en dispositivos de almacenamiento masivo

ABSTRACT

This paper presents an investigation to explicit content images detection on mass storage devices.

This work is based mainly on skin detection method; this requires the use of different techniques, as pattern recognition in the color model HSV or YCbCr.

Moreover a threshold to classify pixels with some skin color tone for both color models using in this work is proposed, this to generate a classifier which can discriminate all objects within the analyzed image.

The proposed system was compared with two commercial systems which carry out with the same goal as that is presented in this thesis. Finally can be probe that the proposed system be able to detect the images under study better than two software solutions more using for forensic researchers for this reason the proposed method can be applied to computer forensics or in detection of pornographic images stored on mass storage devices.

CONTENIDO

DEDICATORIA.....	i
AGRADECIMIENTOS	ii
RESUMEN	iii
ABSTRACT	iv
CONTENIDO.....	v
ÍNDICE DE FIGURAS	viii
LISTA DE TABLAS.....	xi
CAPÍTULO 1: INTRODUCCIÓN	1
1.1 Planteamiento del Problema	2
1.2 Justificación.....	4
1.3 Objetivos	5
1.3.1 Objetivo General.....	5
1.3.2 Objetivos Específicos	5
1.4 Estado del Arte.....	6
CAPÍTULO 2: MARCO TEÓRICO	9
2.1 Modelos de Color	10
2.1.1 RGB.....	10
2.1.2 RGB Normalizado.....	12
2.1.3 HSV	12
2.1.4 HSI.....	13
2.1.5 TSL.....	15
2.1.6 YCbCr.....	16
2.1.7 YUV	17
2.1.8 YIQ	18
2.1.9 CMY y CMYK.....	18
2.2 Modelado de color de la Piel.....	20

2.2.1	Región de Piel Definida Explícitamente.....	20
2.2.2	Modelado de distribución de piel no paramétricos.....	21
2.2.3	Modelado de distribución de piel Paramétricos	24
2.3	Herramientas de detección de imágenes Pornográficas.....	26
2.3.1	Forensic Toolkit 3.1	26
2.3.2	Paraben's Porn Detection Stick.....	34
2.3.3	Otras Soluciones	38
CAPÍTULO 3: ANÁLISIS DE IMÁGENES EXPLÍCITAS		39
3.1	Introducción.....	40
3.2	Diagrama del Sistema Propuesto.....	41
3.2.1	Módulo del cálculo del umbral	41
3.2.2	Módulo de Prueba	41
3.3	Metodología	42
3.3.1	Módulo de Cálculo del umbral	42
3.3.2	Módulo de Prueba	51
3.3.3	Diseño de la Interfaz.....	59
CAPÍTULO 4: RESULTADOS EXPERIMENTALES		63
4.1	Introducción.....	64
4.2	Sistema Propuesto.....	64
4.2.1	Modelo HSV.....	64
4.2.2	Modelo YCbCr	67
4.3	FTK 3.1 Explicit Image Detection	71
4.4	Paraben's Porn Detection Stick.....	80
CONCLUSIONES		89
5.1	Conclusiones Generales	90
5.2	Logros del proyecto.....	91
5.3	Trabajo a Futuro.....	92
REFERENCIAS.....		93
GLOSARIO.....		97
ANEXO A: PROGRAMAS		101

ANEXO B: PUBLICACIONES 135

ÍNDICE DE FIGURAS

Capítulo 1: Introducción

Fig. 1.1 Usuarios de Internet en México	2
---	---

Capítulo 2: Marco Teórico

Fig. 2.1 Modelo de Color RGB	10
Fig. 2.2 Representación Modelo de Color RGB	11
Fig. 2.3 Modelo de Color HSV	12
Fig. 2.4 Modelo de Color HSI	14
Fig. 2.5 Modelo de Color CMY y CMYK	19
Fig. 2.6 Arquitectura de Red de Kohonen	23
Fig. 2.7 Creación de un nuevo usuario	27
Fig. 2.8 Creación de un nuevo caso	28
Fig. 2.9 Parámetros del análisis	28
Fig. 2.10 Algoritmos de FTK para análisis de imágenes	30
Fig. 2.11 Interfaz Paraben's Porn Detection Stick	36
Fig. 2.12 Configuración Paraben's Porn Detection Stick	36
Fig. 2.13 Reporte de imágenes detectadas	37

Capítulo 3: Análisis de Imágenes Explícitas

Fig. 3.1 Módulo Cálculo del Umbral	41
Fig. 3. 2 Módulo de prueba del sistema propuesto	42
Fig. 3.3 Pre-procesamiento imagen	43
Fig. 3.4 Rostro de una persona con piel oscura	43
Fig. 3.5 Rostro de una persona con piel blanca	44
Fig. 3.6 Componentes de imagen <i>HSV</i>	45
Fig. 3.7 Histogramas de la componente <i>H</i>	45
Fig. 3.8 Histogramas de la componente <i>S</i>	46
Fig. 3.9 Histogramas de la componente <i>V</i>	47
Fig.3. 10 Detección de Piel usando el umbral del modelo <i>HSV</i>	47
Fig. 3.11 Imagen Transformada y sus componentes Modelo <i>YCbCr</i>	48
Fig. 3.12 Histogramas de la componente <i>Y</i>	49

Fig. 3.13 Histogramas de la componente Cb	49
Fig. 3.14 Histogramas de la componente Cr	50
Fig. 3.15 Detección de Piel con Umbral del modelo YCbCr	51
Fig. 3.16 Transformación de una imagen de un paisaje.....	52
Fig. 3. 17 Transformación de una imagen de una persona	52
Fig. 3.18 Transformación de una imagen.....	53
Fig. 3.19 Transformación de una imagen de personas	53
Fig. 3.20 Imagen de un paisaje filtrada	54
Fig. 3.21 Imagen filtrada de una persona.....	54
Fig. 3.22 Imagen Filtrada	55
Fig. 3.23 Imagen Filtrada de dos personas	55
Fig. 3.24 Interfaz Desarrollada	59
Fig. 3.25 Interfaz para análisis de una imagen.....	60
Fig. 3.26 Análisis de una imagen sin contenido Explícito	60
Fig. 3.27 Análisis de una imagen con contenido Explícito.....	61
Fig. 3.28 Interfaz para analizar un directorio	62
Fig. 3.29 Análisis de un directorio	62

Capítulo 4: Resultados Experimentales

Fig. 4.1 Resultados de la primera prueba	65
Fig. 4.2 Resultados del Segundo Conjunto de Imágenes.....	65
Fig. 4.3 Resultados de la última prueba	66
Fig. 4.4 Porcentajes de Reconocimiento Sistema Propuesto HSV	67
Fig. 4.5 Resultado de Reconocimiento del Conjunto 1 con YCbCr.....	68
Fig. 4.6 Resultado de detección del segundo conjunto.....	68
Fig. 4.7 Resultado del tercer conjunto de pruebas	69
Fig. 4.8 Porcentajes de Reconocimiento, Falsos Positivos y Falsos Negativos con Modelo de Color YCbCr	70
Fig. 4.9 Ventana de adquisición de la evidencia	71
Fig. 4. 10 Primer análisis usando FTK 3.1	72
Fig. 4.11 Ventana del análisis adicional.....	73
Fig. 4.12 Análisis de las imágenes	74
Fig. 4.13 Filtro para el primer y segundo algoritmo.....	75
Fig. 4.14 Filtro para el segundo y tercer algoritmo.....	75
Fig. 4.15 Filtro para el primer y tercer algoritmo.....	75
Fig. 4.16 Administración de los Filtros creados.....	76
Fig. 4.17 Imágenes Filtradas	77
Fig. 4.18 Creación de un marcador	78

Fig. 4.19 Resultados obtenidos con FTK 3.1	79
Fig. 4.20 Porcentajes de reconocimiento, falsos positivos y negativos del primer conjunto usando Paraben's Porn Detection Stick	80
Fig. 4.21 Reporte del primer conjunto usando un valor de sensibilidad de 60	81
Fig. 4.22 Porcentajes de reconocimiento, falsos positivos y negativos del segundo conjunto de imágenes.....	82
Fig. 4.23 Reporte del segundo conjunto de imágenes	83
Fig. 4.24 Resultados del tercer conjunto de imágenes	84
Fig. 4.25 Reporte generado del tercer conjunto de imágenes	85
Fig. 4.26 Porcentaje de Reconocimiento.....	86
Fig. 4. 27 Porcentaje de Falsos Positivos	86
Fig. 4. 28 Porcentaje de Falsos Negativos.....	87

LISTA DE TABLAS

Tabla I Opciones del Procesamiento de la evidencia	29
Tabla II Descripción de los Algoritmos utilizados por EID	30
Tabla III Valores de la Clasificación de imágenes	32
Tabla IV Porcentaje de Piel de Imágenes	56
Tabla V Clasificación de imágenes	58

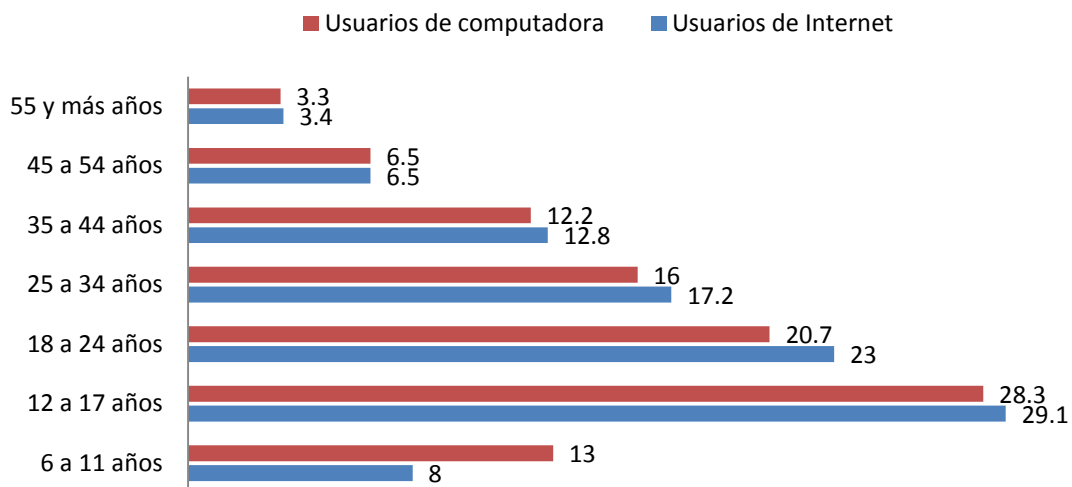
CAPÍTULO 1: INTRODUCCIÓN

En el presente capítulo se presentan los motivos por los que se llevó a cabo esta investigación, las metas propuestas para el cumplimiento de este trabajo, así como una breve descripción de trabajos similares al propuesto en esta tesis.

1.1 Planteamiento del Problema

Con el rápido crecimiento de la red de redes o World Wide Web y el fácil acceso a una computadora con conexión a Internet, las personas tienen cada vez más opciones para poder acceder a estos tipos de medios para buscar información. Mucha de la información que existe o está disponible en Internet son imágenes, videos, etc. En México los niños de 6 a 11 años de edad, así como los jóvenes entre 12 y 17 años de edad, son los usuarios más comunes de estos medios [1], ya que son más del 37% del total de los usuarios como se observa en la Figura 1.1, por lo que están expuestos a observar con mayor frecuencia imágenes clasificadas como para adultos. Por esta razón existe una gran necesidad de desarrollar aplicaciones que puedan detectar este tipo de imágenes antes de que los usuarios puedan verlas.

Usuario de TIC por grupos de edad 2009 (Porcentajes)



Fuente: INEGI. Encuesta Nacional sobre Disponibilidad y Uso de las Tecnologías de la Información en los Hogares, 2009.

Fig. 1.1 Usuarios de Internet en México

Otro factor importante a considerar es que México es primer lugar en América Latina en producción de Pornografía Infantil [2], por lo que también existe la necesidad de desarrollar aplicaciones que permitan detectar imágenes con contenido explícito en las computadoras o dispositivos de almacenamiento, de las personas que producen este tipo de delito.

Es importante desarrollar aplicaciones que puedan detectar imágenes con algún contenido pornográfico, en computadoras y dispositivos de almacenamiento como: discos duros, memorias USB, etc., porque las aplicaciones que existen actualmente, tratan de evitar sólo uno de los problemas existentes, bloqueando los sitios de Internet que los usuarios, en este caso son los niños y jóvenes, quienes navegan en Internet en busca de entretenimiento o información importante para ellos.

Existen varias aplicaciones que bloquean los sitios de Internet que contienen contenido pornográfico, entre las existentes están las siguientes: CyberPatrol [3], ContentProtect [4], NetNanny [5], Family.net [6] y K9 Web Protection [7]. En México sólo se ha desarrollado una aplicación llamada Saint Lite 2.0 [8] [9], la cual filtra el contenido de las páginas de Internet visitadas por los usuarios.

El otro problema es detectar las imágenes una vez que ya están almacenadas en una computadora o dispositivo de almacenamiento, para resolver este problema existen pocas soluciones en el mercado, dentro de las más comunes están las que ofrecen las empresas AccessData y Paraben, las cuales tienen como oferta Forensic Toolkit 3.1 Explicit Image Detection y Paraben's Porn Detection Stick, respectivamente. Estas soluciones son aplicadas a la Informática Forense.

1.2 Justificación

Los navegadores de Internet utilizan filtros para controlar que los menores de edad puedan observar contenido explícito mientras están navegando en Internet, existen aplicaciones comerciales que hacen que los padres de familia puedan bloquear sitios con contenido explícito y de esta manera tener un mayor control sobre el contenido que buscan sus hijos en Internet.

Por otro lado son pocas las opciones que existen para poder realizar un análisis cuando imágenes con contenido explícito se encuentran dentro de un dispositivo de almacenamiento, y estas aplicaciones tienen un costo elevado en el mercado, así como su uso se limita a un lapso de tiempo limitado.

1.3 Objetivos

1.3.1 Objetivo General

- Diseñar e implementar un algoritmo capaz de detectar imágenes con contenido explícito, utilizando una metodología propia y compararlo con al menos dos sistemas propietarios de detección imágenes con contenido explícito.

1.3.2 Objetivos Específicos

- Aprender a utilizar las aplicaciones y herramientas, Forensic Toolkit 3.1 Explicit Image Detection y Paraben's Porn Detection Stick.

- Diseñar e Implementar un algoritmo capaz de detectar imágenes con contenido explícito.

- Diseñar una interfaz gráfica que implemente el algoritmo propuesto.

1.4 Estado del Arte

En la actualidad se han hecho diversas investigaciones que tratan el mismo problema al que se propone en este trabajo.

La primera investigación fue hecha en el año 1996 por M. Fleck *et al.* [10] quienes crearon un sistema capaz de identificar imágenes a color de personas desnudas, usando la detección de piel y la geometría del cuerpo humano para detectar a las personas existentes dentro de la imagen. Después de esta investigación se derivaron muchos trabajos tratando de resolver el mismo problema.

Algunos investigadores han tratado de resolver el problema de detección de piel creando un umbral para diferentes modelos de color, en el cual puedan clasificar los píxeles de piel y no piel, pero ninguno de éstos han trabajado en la detección de imágenes con contenido explícito.

Dai y Nakano [11] en 1996 propusieron un sistema para detección de rostros usando un modelo de detección de piel definido explícitamente para el modelo de color YIQ , ellos toman solamente el rango de valores de la componente I para determinar las zonas de piel de personas asiáticas, el rango para I es de $[0, 50]$.

Sobotkka y Pitas [12] utilizaron un rango de valores para el modelo de color HSV , pero sólo de las componentes HS , para la detección de rostros. El rango de valores para H es $[0, 50]$ y para S de $[0.23, 0.68]$. Este intervalo sirve para el reconocimiento de imágenes de personas con color de piel blanca y amarilla. Chai y Ngan [13] propusieron un algoritmo para la segmentación de rostros, el cual usaron un límite para las componentes Cb y Cr del modelo de color $YCbCr$. Los valores de los píxeles deberían de estar

en el rango, $Cb = [77, 127]$ y $Cr = [133, 173]$, para poder considerarse como piel.

Garcia y Tziritas [14] presentaron en su trabajo para la detección del rostro, segmentando la piel usando ocho planos en el modelo de color $YCbCr$ o en seis planos en el modelo de color HSV . Wang y Yuan [15] utilizaron un umbral para la combinación de los modelos de color RGB y HSV . Los valores del umbral es el siguiente: $R = [0.36, 0.465]$, $G = [0.28, 0.363]$, $H = [0, 50]$, $S = [0.20, 0.68]$, y $V = [0.35, 1]$, para discriminar entre píxeles de color piel y no piel. Yao y Gao [16] primero transforman del modelo de color RGB al YUV y YIQ para usar la crominancia como componente para proponer un rango de valores para la detección de piel y de los labios. Wong *et al.* [17] utiliza un intervalo para la luminancia Y , en el modelo de color $YCbCr$. Tomaz *et al.* [18] usa un umbral para las componentes TS , en el modelo de color TSL .

Existen otros investigadores que utilizan otros métodos de detección de piel, pero trabajan con los mismos modelos de color, que en este trabajo se plantea. En el modelo de color HSV , R. Schettini *et al.* [19] diseñaron un algoritmo que distinguiera entre imágenes pornográficas y no, para los filtros de web semántica, usando arboles de decisión a través de una metodología de Arboles de clasificación y Regresión (Classification And Regression Tress, CART), y la Máquina de Soporte de Vectores (Support Vectors Machine, SVM). Yi Chan *et al.* [20] hicieron una propuesta para crear un sistema capaz de bloquear la pornografía en los navegadores web. A. Bosson *et al.* [21] hicieron una comparación entre los diferentes clasificadores como: Máquina de Soporte de Vectores (Support Vectors Machine, SVM), Modelo Lineal Generalizado (Generalized Linear Model, GLM), Perceptrón Multicapa (Multilayer Perceptron, MLP), y K -Vecinos Próximos (K Nearest Neighbor, k-Nn), usando datos estadísticos obtenidos de la detección de piel. W. Kim *et al.* [22] clasifica las imágenes a través de la obtención de siete descriptores que obtiene de cada imagen, usando MPEG-7. Q. Zhu *et al.* [23] utiliza en su

trabajo un modelo de mezclas gaussianas (Gaussian Mixture Model, GMM) y SVM, para la clasificación de imágenes pornográficas.

Para el modelo de color $YCbCr$, W.Arentz y B.Olstad [24] propusieron un algoritmo para la detección de sitios web con contenido pornográfico, a través del análisis de las imágenes existentes en ese sitio. QF.Zheng *et al.* [25] combinan el reconocimiento de rostro y la detección de piel, así como la cantidad de rostros para conocer si la imagen analizada es una imagen pornográfica. J.-S. Lee *et al.* [26] usan la detección de piel y el análisis de textura para saber el contenido de la imagen.

W. Kelly *et al.* [27] presenta en su trabajo una revisión de las técnicas utilizadas para detectar la piel humana que ocupan los diferentes autores en sus investigaciones publicadas. P. Kakumanu *et al.* [28] hacen una revisión de técnicas de detección piel usadas por diferentes autores, así como los clasificadores utilizados en dichos trabajos, además de que hacen una comparación entre cada uno de éstos.

CAPÍTULO 2: MARCO TEÓRICO

En este capítulo se verán los modelos de color existentes para poder realizar el procesamiento digital de imágenes, así como sus formulas matemáticas para poder transformar una imagen de un modelo de color a otro. También se conocerán cuales son las técnicas más comunes para poder generar un modelo capaz de clasificar los píxeles como piel o no. Además se dará una introducción a las herramientas utilizadas para el análisis de imágenes con contenido explícito.

2.1 Modelos de Color

Un modelo de color, o también llamado espacio de color en diferentes bibliografías, se refiere a una representación matemática de un conjunto de colores. Estos modelos se pueden dividir en cuatro categorías: Modelos de color Básicos (*RGB*, *RGB Normalizado*), Modelos de color Perceptuales (*HSV*, *HSI*, *TSL*), Modelos de color Ortogonales (*YCbCr*, *YUV*, *YIQ*) y modelos de color (*CMY*, *CMYK*) [28]. En esta sección se explicará brevemente cada uno de los modelos de color antes mencionados.

2.1.1 RGB

El modelo de color RGB es un modelo aditivo en el cual los colores primarios rojo, verde y azul son sumados en varias formas para reproducir diferentes colores. El nombre viene de las iniciales en inglés de los tres colores (Red, Green y Blue). Una representación del modelo es mostrado en la Figura 2.1.

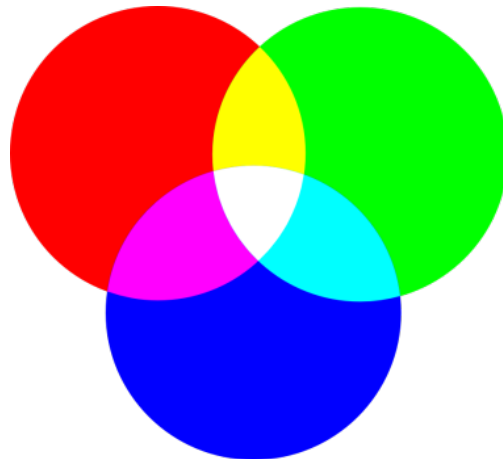


Fig.2.1 Modelo de Color RGB

Este modelo está basado en el sistema de coordenadas cartesiano. El subespacio de color de interés es el cubo mostrado en la Figura 2.2, en el cual los valores: rojo, verde y azul están en tres esquinas; cian, magenta y amarillo en otras tres esquinas; el negro es el origen; y el blanco es la esquina más alejada del origen. En este modelo, la escala de grises va desde el negro al blanco a lo largo de estos dos puntos [29].

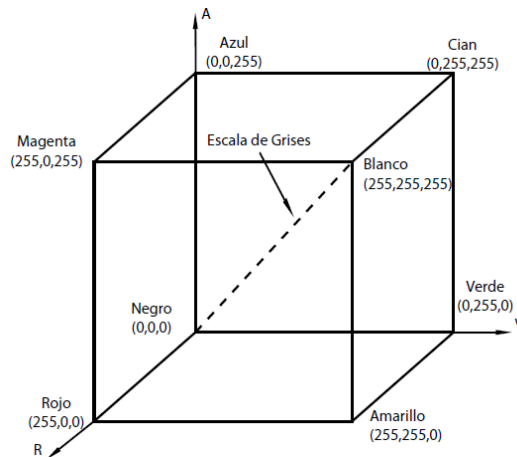


Fig.2.2 Representación Modelo de Color RGB

Un color en el modelo RGB es descrito indicando el valor de cada componente roja, verde y azul, el cual puede variar desde 0 a un valor máximo que depende de la aplicación. En computación los valores de las componentes son almacenados como números enteros en el rango de 0 a 255.

2.1.2 RGB Normalizado

El modelo RGB normalizado es una representación, que se puede obtener de una manera sencilla usando los valores RGB por el siguiente procedimiento de normalización mostrado en (1). La normalización es un método para evitar problemas de los cambios de iluminación en una imagen.

$$r = \frac{R}{R+G+B}; \quad g = \frac{G}{R+G+B}; \quad b = \frac{B}{R+G+B} \quad (1)$$

De esta manera la suma de las tres componentes normalizadas debe de ser $(r + g + b = 1)$ [28].

2.1.3 HSV

Este modelo de color es considerado más cercano que el modelo de color RGB en la forma en que las personas describen las sensaciones de color. El modelo de color HSV obtiene el nombre de las siglas en inglés (Hue, Saturation, Value) que en español significan Tonalidad, Saturación, Valor, respectivamente; es una transformación no lineal del modelo de color RGB, y los colores son una combinación de los tres valores: Tonalidad, Saturación o cantidad de color, y su valor. Estos valores son representados en un diagrama circular, que es mostrado en la Figura 2.3.

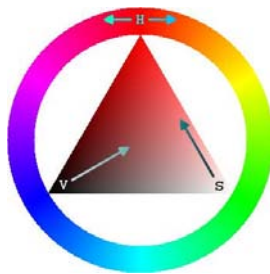


Fig.2.3 Modelo de Color HSV

Las tres magnitudes pueden tener los siguientes valores:

- Tonalidad: Es el tipo de color (por ejemplo: rojo, verde, o amarillo), que son representados como el valor del grado de un ángulo, cuyos valores posibles están en el rango de 0° a 360° (aunque para algunas aplicaciones estos valores son normalizados de 0% a 100%).
- Saturación: Es representado como la distancia del eje del brillo negro – blanco. Los posibles valores están en el rango de 0% a 100%.
- Valor: Representa la altura en el eje negro – blanco. Los posibles valores están en el rango de 0% a 100%. Donde 0 es siempre negro, dependiendo de la saturación, 100 podría ser blanco o un color más o menos saturado.

Para convertir del modelo de color RGB al HSV se usan las expresiones matemáticas (2) al (4), que son mostradas a continuación:

$$H = \arccos \frac{\frac{1}{2}[(R-G) + (R-B)]}{\sqrt{[(R-G)^2 + (R-B)(G-B)]}} \quad (2)$$

$$S = 1 - 3 \frac{\min(R, G, B)}{R + G + B} \quad (3)$$

$$V = \frac{1}{3}(R + G + B) \quad (4)$$

2.1.4 HSI

El modelo de color HSI obtiene el nombre de las siglas en inglés (Hue, Saturation, Intensity) que en español significan Tonalidad, Saturación, Intensidad, respectivamente. Este modelo es más adecuado que el modelo de color RGB para algunas tareas de procesamiento de imágenes [29].

Donde H especifica el color puro dominante percibido por una persona (ejemplo: azul, rojo, verde), S mide el ángulo en el que el color puro ha sido diluido por la luz blanca. Ya que el color y la intensidad son independientes, se puede manipular uno sin afectar al otro.

El modelo de color HSI es descrito en un sistema de coordenadas cilíndricas y es representado comúnmente como un doble cono, como se muestra en la Figura 2.4. Un color es un sólo punto dentro o en la superficie del doble cono. La altura del punto corresponde a la intensidad y la saturación es la longitud del vector de intensidad y el tono o H es su orientación, expresado como un ángulo en grados.

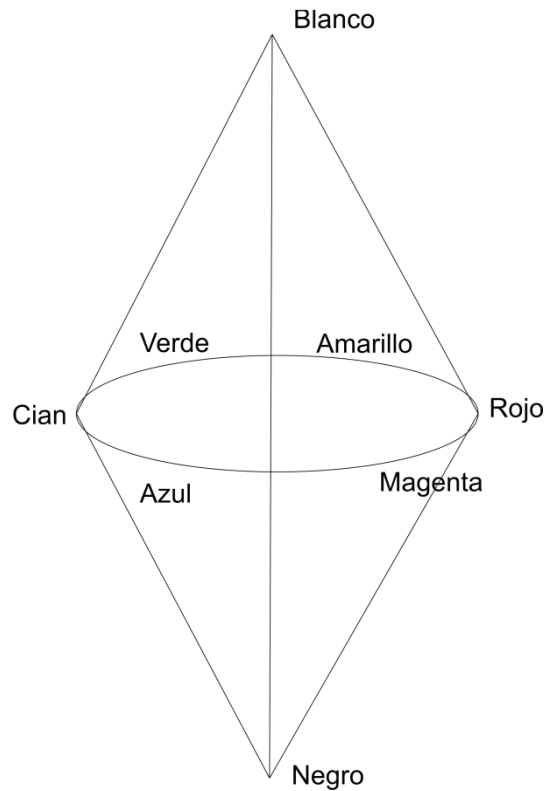


Fig.2.4 Modelo de Color HSI

Para convertir del modelo de color RGB a HSI, se utilizan las expresiones matemáticas (5) a (8).

$$H = \begin{cases} \theta & \text{sí } B \leq G \\ 360 - \theta & \text{sí } B > G \end{cases} \quad (5)$$

$$\theta = \cos^{-1} \frac{\frac{1}{2}[(R-G) + (R-B)]}{\left[(R-G)^2 + (R-B)(G-B) \right]^{1/2}} \quad (6)$$

$$S = I - 3 \frac{\min(R, G, B)}{R + G + B} \quad (7)$$

$$I = \frac{I}{3}(R + G + B) \quad (8)$$

2.1.5 TSL

El modelo de color TSL obtiene el nombre de sus siglas en inglés de (Tint, Saturation, y Lightness), que en español es Tinte, Saturación y Luminosidad. Es un modelo de color normalizado de crominancia-luminancia, donde el modelo de color es una transformación del modelo de color RGB normalizado, para lograr la transformación se ocupan las expresiones matemáticas (9) a (11) que se muestran a continuación:

$$T = \begin{cases} \frac{\arctan\left(\frac{r'}{g'}\right)}{2\pi} + \frac{1}{4}; & g' > 0 \\ \frac{\arctan\left(\frac{r'}{g'}\right)}{2\pi} + \frac{3}{4}; & g' < 0 \\ 0; & g' = 0 \end{cases} \quad (9)$$

$$S = \sqrt{\frac{9}{5}(r'^2 + g'^2)} \quad (10)$$

$$L = 0.299R + 0.587G + 0.114B \quad (11)$$

Donde:

$$r' = r - \frac{1}{3}, \quad g' = g - \frac{1}{3}, \text{ y } r, g \text{ se obtienen de (1).}$$

2.1.6 YCbCr

En este formato, la información de la luminancia está representada por una sola componente, Y , y la información de color es almacenada como dos componentes de la diferencia de color, Cb y Cr . La componente de crominancia azul o Cb es la diferencia entre la componente azul y el valor de referencia, y la componente de crominancia roja o Cr es la diferencia entre la componente roja y el valor de referencia. La matriz utilizada para convertir el modelo de color RGB al YCbCr es (12):

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.000 \\ 112.000 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (12)$$

2.1.7 YUV

El modelo de color YUV es usado en los sistemas PAL (Phase Alternation Line) y NTSC (National Television System Committee) de difusión televisiva, los cuales son los estándares en la mayoría de los países del mundo. YUV está definido [32] en términos de una componente de luminancia (Y) y dos componentes de crominancia (U , V). Codifica una imagen o video color teniendo en cuenta la percepción humana, permitiendo usar un ancho de banda menor para las componentes de crominancia (U , V), disminuyendo los errores de transmisión o imperfecciones de compresión, que usando una representación RGB .

La matriz que se utiliza para hacer la transformación es (13), que fue obtenida de:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (13)$$

Los valores para RGB están en el rango de 0-255, después de la transformación, Y tiene un rango de 0-255, U un rango de 0 a ± 112 , y V un rango de 0 a ± 157 .

2.1.8 YIQ

El modelo de color *YIQ* se deriva del modelo de color *YUV* [32] y es usado opcionalmente por el estándar de vídeo de color *NTSC*. Las siglas *I* y *Q*, en fase y cuadratura, respectivamente. La cuadratura es el método de modulación usado para la transmisión de información de color. La matriz para convertir entre el modelo de color *RGB* y *YIQ* es (14):

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.257 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (14)$$

Los valores para *RGB* están en el rango de 0-255, después de la conversión, *Y* tiene un rango de 0-255, *I* un rango de 0 a ± 152 , y *Q* un rango de 0 a ± 134 .

2.1.9 CMY y CMYK

Cian, Magenta y Amarillo son los colores secundarios, o alternativos de los colores primarios. Por ejemplo cuando una superficie cubierta con color cian es iluminada luz blanca, la luz roja no es reflejada en la superficie. Esto es, el color cian sustrae la luz roja de la luz blanca reflejada, la cual está compuesta de igual cantidad de luz roja, verde y azul.

La mayoría de los dispositivos que ponen pigmentos de color en papel como impresoras y copiadoras de color, requieren una entrada de datos *CMY* o realizar una conversión interna de *RGB* a *CMY*. Esta conversión es realizada usando la ecuación (15).

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} I \\ I \\ I \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (15)$$

Donde los valores de RGB deben estar normalizados entre el rango [0,1].

En teoría, la misma cantidad de pigmentos de colores primarios cian, magenta y amarillo podría producir el color negro. Entonces con el fin de producir un negro verdadero (el cual es el color predominante en las impresiones), es agregado un cuarto color, negro, dando lugar al modelo de color *CMYK*.

Estos modelos están representados en la Figura 2.5, que es mostrado a continuación:

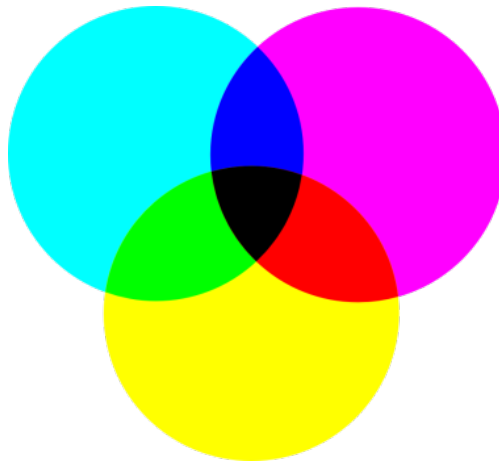


Fig.2.5 Modelo de Color CMY y CMYK

2.2 Modelado de color de la Piel

El objetivo principal de la detección piel es crear un criterio para tomar una decisión, que pueda discriminar entre píxeles con algún tono de piel y píxeles sin algún tono de piel. Esto se logra usualmente introduciendo una métrica, la cual mide el color del pixel. El tipo de métrica es definida por el método del modelo de color de piel.

2.2.1 Región de Piel Definida Explícitamente

Este es uno de los métodos más sencillos de realizar, ya que se define claramente o específicamente el límite de la región de piel, a través de una serie de reglas numéricas. Este método es aplicable en diferentes modelos de color. Peer *et al.* [31] definió el siguiente límite para el modelo de color RGB, que se muestra en (16).

$$\begin{aligned} & \text{Un píxel es clasificado como piel, sí :} \\ & R > 95 \text{ y } G > 40 \text{ y } B > 20 \text{ y} \\ & \max(R, G, B) - \min(R, G, B) > 15 \text{ y} \\ & |R - G| > 15 \text{ y } R > G \text{ y } R > B \end{aligned} \tag{16}$$

La principal ventaja de este método es la simplicidad de las reglas de la detección de piel, que lleva a la construcción de un clasificador rápido.

2.2.2 Modelado de distribución de piel no paramétricos

La idea principal de los modelados de distribución de piel no paramétricos, es estimar la distribución del color de la piel de los datos de entrenamiento sin crear un modelo de piel explícito. El resultado de estos métodos a veces es referido a la construcción de un mapa de Probabilidad de Piel (Skin Probability Map, SPM), asignando un valor de probabilidad a cada punto del modelo de color.

- **Tabla de Búsqueda Normalizada (LUT)**

En una tabla de búsqueda normalizada, el plano de la crominancia de los modelos de color es cuantificado en una serie de rangos o intervalos, cada uno correspondiente a un determinado rango del valor de la componente de color, en pares en el caso de representaciones en dos dimensiones, o en tríadas en el caso de representaciones en tres dimensiones. Estos intervalos forman un histograma de dos o tres dimensiones (2D o 3D), que son referidos a la tabla de búsqueda normalizada (LUT). Cada intervalo almacena el número de veces que un color en particular aparece en las imágenes de piel de entrenamiento. Después del entrenamiento los histogramas son normalizados, convirtiendo los valores del histograma en una distribución de probabilidad discreta, como se muestra en (17):

$$P_{skin}(c) = \frac{skin[c]}{Norm} \quad (17)$$

Donde:

$skin[c]$: Es el valor del intervalo del histograma correspondiente al vector de color c .

Norm: Es el coeficiente de normalización o la suma de todos los valores del intervalo del histograma según Jones y Regh [34], o el valor máximo del histograma por Zarit *et al.* [35].

Los valores normalizados de la tabla de búsqueda normalizada constituye la probabilidad de que un color corresponderá a un color de piel.

- **Clasificador de Bayes**

Jones y Regh [34] calcularon dos diferentes histogramas, de piel y no piel. Dados los histogramas de piel y no piel, la probabilidad de que un color determinado pertenezca a la clase de piel y no piel, también llamada clase de probabilidades condicionales o regla de Bayes, se define como (18):

$$P(c|skin) = \frac{s(c)}{T_s}, \quad P(c|non - skin) = \frac{n(c)}{T_n} \quad (18)$$

Donde $s(c)$ es el número de píxeles en el color c del intervalo del histograma de piel, $n(c)$ es el número de píxeles en el color c del intervalo del histograma de no piel. T_s y T_n representan el número total en el intervalo de los histogramas de piel y no piel.

Dada la clase de probabilidad condicional de los modelos de piel y no piel, un clasificador puede ser construido usando la máxima verosimilitud (ML) de Bayes. Utilizando esto, un píxel de una imagen puede ser clasificado como piel sí cumple con (19):

$$\frac{P(c|skin)}{P(c|non - skin)} \geq \Theta \quad (19)$$

Donde $0 \leq \Theta \leq 1$ es el valor del umbral, el cual puede ser ajustado entre verdaderos positivos y falsos positivos. Este valor de umbral es normalmente determinado de la curva de ROC (características operativas del receptor), que es calculada del conjunto de datos de entrenamiento. La curva

ROC muestra la relación entre los verdaderos positivos y falsos positivos como función del umbral de detección Θ . El histograma basado en el clasificador de Bayes, también llamado SPM, ha sido usado en general para la segmentación de piel.

- **Mapa de Autoorganización de Kohonen**

El mapa de autoorganización (Self Organizing Map, SOM), también es conocido como red de Kohonen, es uno de los modelos de Redes Neuronales Artificiales más populares. El algoritmo SOM está basado en el aprendizaje sin supervisión.

En este modelo [34], las neuronas se organizan en una arquitectura unidireccional de dos capas, como se muestra en la Figura 2.6. La primera es la capa de entrada o sensorial, que consiste en m neuronas, una por cada variable de entrada, que se comportan como buffers, distribuyendo la información procedente del espacio de entrada a las neuronas de la segunda capa. Las entradas son muestras estadísticas $x(t) \in \mathfrak{R}^m$ del espacio sensorial. El procesamiento se realiza en la segunda capa, que forma el mapa de rasgos, consistente habitualmente en una estructura rectangular de $n_x \times n_y$ neuronas que operan en paralelo.

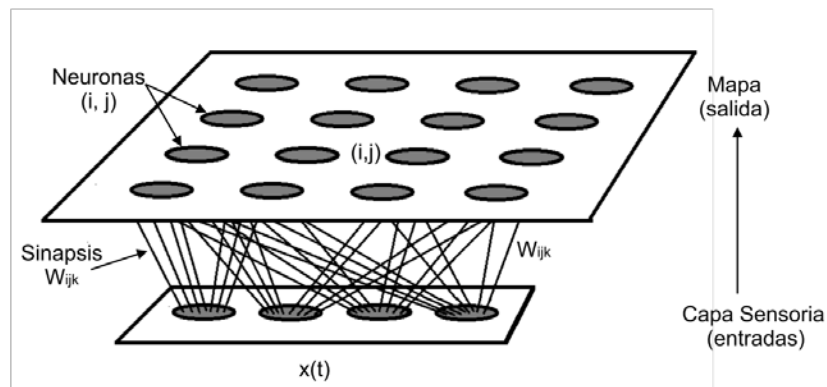


Fig.2.6 Arquitectura de Red de Kohonen

El mapa puede describirse como una matriz de procesadores elementales (i, j) ordenados en dos dimensiones, que almacenan un vector de pesos sinápticos o vector de referencia $w_{ij}(t)$, con $\{w_{ij}(t); w_{ij} \in \mathfrak{R}^m, 1 \leq i \leq nx, 1 \leq j \leq ny\}$.

2.2.3 Modelado de distribución de piel Paramétricos

Los modelados de distribución de piel paramétricos están basados en reglas de decisión fijas, las cuales están definidas empíricamente en distintos modelos de color después del análisis de distribución de piel. Estas reglas son aplicadas para determinar si un píxel de color a través de su valor es piel o no.

- **Gausiana Simple**

La distribución del color de la piel puede ser modelada por una gaussiana elíptica de probabilidad conjunto una función de densidad de probabilidad, la cual está definida en (20):

$$P(c) = \frac{1}{[2\pi|\Sigma|]^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(c - \mu)^T \Sigma^{-1}(c - \mu)\right] \quad (20)$$

Donde c es el vector de color, μ y Σ son el vector de la media y la matriz de covarianza, respectivamente. La probabilidad $P(c)$ es usada para calcular la probabilidad del color de la piel.

- **Mezcla de Gaussianas**

Es un modelo más sofisticado capaz de describir distribuciones de forma compleja. Una función de densidad de mezcla de Gaussianas es expresada como (21):

$$P(c) = \sum_{i=1}^N w_i \frac{1}{[2\pi|\Sigma|]^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(c-\mu)^T \Sigma^{-1}(c-\mu)\right] \quad (21)$$

Donde N es el número de componentes de la mezcla y el factor ponderado, w_i es la contribución de la i -ésima Gaussiana. El entrenamiento de este modelo se efectúa a través de la iteración de la técnica de maximización de expectativa (EM), la cual garantiza que la componente N sea conocida de antemano. Una parte crítica de este método es la selección del número de la componente N .

- **Modelo de frontera elíptica**

El modelo de frontera elíptica se presenta como una alternativa para resolver la alta tasa de positivos encontrados en el modelo de Gaussiano Simétrico. La realización de este método es comparable a la Mezcla de Gaussianas y es tan simple como una Gaussiana Simple en complejidad computacional. Este modelo se describe como (22).

$$\Phi(c) = [c - \varphi]^T \wedge^{-1} [c - \varphi] \quad (22)$$

Donde c es el vector de color, φ y \wedge son los parámetros del modelo definido como (23).

$$\varphi = \frac{1}{n} \sum_{i=1}^n c_i, \quad \wedge = \frac{1}{N} \sum_{i=1}^n f_i (c_i - \mu)(c_i - \mu)^T \quad (23)$$

Donde N es el número total de muestras en el conjunto de entrenamiento, f_i es el número de muestras con crominancia, mientras μ es la media del vector de crominancia del conjunto de entrenamiento. Un píxel con crominancia c es clasificado como piel sí $\Phi(c) < \phi$, donde ϕ es el valor del umbral seleccionado como compensación entre verdaderos y falsos positivos.

2.3 Herramientas de detección de imágenes Pornográficas

2.3.1 Forensic Toolkit 3.1

Forensic Toolkit (FTK) 3.1 [35] es una solución en software que ofrece la empresa Access Data para la informática forense. FTK proporciona la tecnología que es utilizada por profesionales de seguridad en informática para filtrar, analizar, investigar e informar sobre los resultados de las pruebas.

FTK provee a los usuarios la capacidad de realizar inspecciones forenses exhaustivas a las computadoras. FTK puede ser usado para adquirir, preservar, analizar y presentar evidencia digital.

Todos los análisis de informática forense requieren de los siguientes pasos:

1. Adquirir: Identificar y asegurar la evidencia.
2. Preservar: Crear y almacenar una imagen forense de la evidencia.
3. Analizar: Crear un caso en algún programa que proporcione las herramientas necesarias para investigar adecuadamente las pruebas.
4. Presentar: Crear un informe del caso para documentar y sintetizar los resultados de la investigación.
5. Administrar: Crear una copia de seguridad, restaurar y eliminar los casos y las pruebas.

La evidencia en un caso de informática forense depende en gran medida de la protección de los datos y la adquisición de una manera que garantice la integridad de la información.

Para realizar la adquisición de la evidencia se crea una imagen forense usando la herramienta FTK Imager, que es una herramienta de adquisición, con la cual se crea una imagen forense del dispositivo de una manera rápida y garantiza la evidencia para futuros análisis forenses. Esta herramienta hace un duplicado del medio o dispositivo bit a bit, esto quiere decir que se crea una copia idéntica al dispositivo obteniendo los archivos, espacio libre, errores que pueda tener dicho dispositivo, así como la información borrada.

Existen 3 tipos de evidencia que se pueden obtener usando FTK Imager, las cuales son las siguientes: La evidencia que fue adquirida antes de ser añadida a algún caso en FTK se le conoce como **evidencia estática**, la evidencia que se obtiene cuando el equipo o computadora está en funcionamiento se le conoce como **evidencia en vivo**. Otro tipo de adquisición es cuando se obtiene la evidencia de alguna computadora que se encuentra en la misma red de datos que la computadora desde donde se analiza el caso, a esta tipo de evidencia se le conoce como **evidencia remota**.

Para poder comenzar a trabajar con FTK, el primer paso después de que el software fue instalado en la computadora, es crear un usuario como Administrador con su respectiva contraseña, como se muestra en la Figura 2.7.



Fig.2.7 Creación de un nuevo usuario

Posteriormente se crea un nuevo caso, donde algunos campos deben ser llenados para poder identificar el caso después de algún tiempo de manera sencilla, la ventana para crear un caso se muestra en la Figura 2.8.

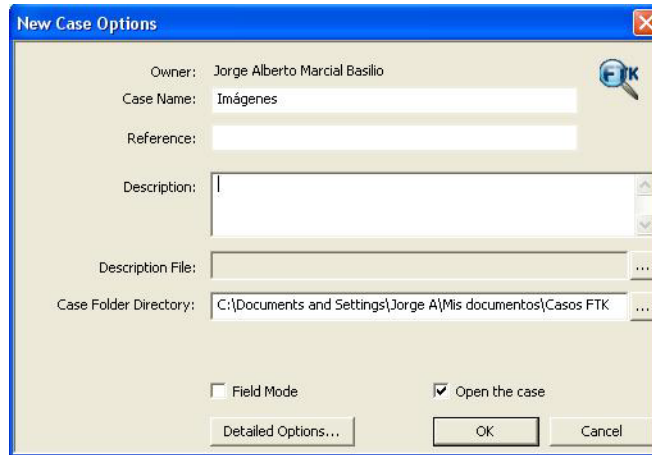


Fig.2.8 Creación de un nuevo caso

Para obtener la información deseada en la búsqueda durante el análisis de imágenes con contenido explícito, se seleccionan los parámetros mostrados en la Figura 2.9.

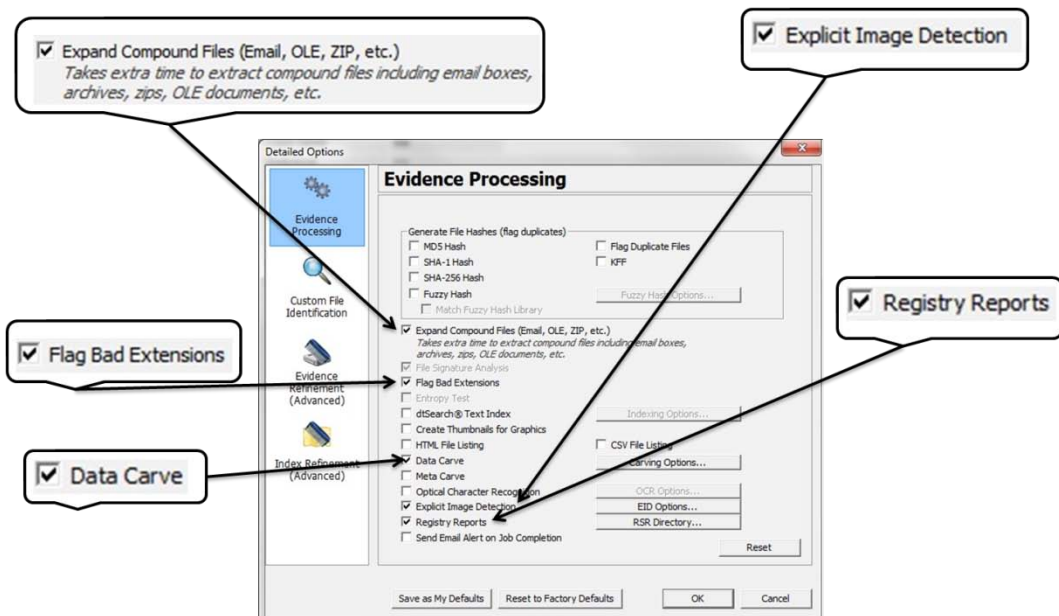


Fig.2.9 Parámetros del análisis

En la Tabla I se muestra la descripción de cada uno de los parámetros seleccionados para la realización del análisis.

Tabla I Opciones del Procesamiento de la evidencia

Nombre del Proceso	Descripción
Expand Compound Files	Abre automáticamente y procesa el contenido de los archivos compuestos como: *.ZIP, email y *.OLE.
File Signature Analysis	Analiza los archivos que indiquen si sus cabeceras o las firmas coinciden con sus extensiones. Esta opción está predeterminada.
Flag Bad Extensions	Identifica los archivos que no coinciden con su extensión, basándose en la información de la cabecera.
Data Carve	Utiliza la firma de los archivos para identificar los archivos eliminados en la evidencia.
Explicit Image Detection	Los algoritmos del modulo EID permiten analizar el contenido de una imagen, para saber si es o no una imagen con contenido explícito.
Registry Reports	Crea un reporte del contenido del caso automáticamente.

Para el caso de estudio en esta tesis la opción más importante es la de Explicit Image Detection (EID) que es la opción que permite analizar el contenido explícito de una imagen, existen tres algoritmos para examinar las imágenes, los cuales se muestran en la Figura 2.10.

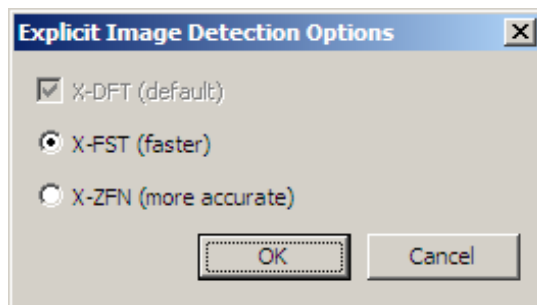


Fig.2.10 Algoritmos de FTK para análisis de imágenes

La descripción de cada algoritmo se muestra en la Tabla II.

Tabla II Descripción de los Algoritmos utilizados por EID

Nombre	Nivel	Descripción
X-DFT	XD1 (Default)	Este algoritmo es generalmente el más preciso, además de que es el que selecciona automáticamente el software.

Tabla II (cont.) Descripción de los Algoritmos utilizados por EID

Nombre	Nivel	Descripción
X-FST	XTB (Rápido)	<p>Este algoritmo es el más rápido de los tres. Le asigna valores numéricos a un folder o directorio dependiendo del número de archivos que éste contenga, pero sólo de los que cumplan con el criterio de alta probabilidad de ser material explícito.</p> <p>Está basado en una tecnología diferente a la del algoritmo X-DFT y no usa regularmente DNA's. Está diseñado para una gran cantidad de imágenes, su propósito es reducir rápidamente o filtrar significativamente el número de imágenes.</p>
X-ZFN	Menor Falsos Negativos (XT2)	<p>Este algoritmo es similar al S-FST pero con más características y con menos falsos negativos que X-DFT.</p> <p>Se aplica este algoritmo después del análisis inicial de toda la evidencia o sólo para los folders marcados como altamente por el algoritmo X-FST.</p>

Se necesita hacer dos análisis, porque el primero sólo puede evaluar a las imágenes usando dos de los algoritmos que se mostraron anteriormente, al finalizar esta primera investigación se hace un análisis adicional para poder utilizar el otro algoritmo, y de esta forma tener un mejor resultado a la hora de clasificar las imágenes con contenido explícito.

Los algoritmos utilizados por FTK 3.1 clasifican las imágenes usando los valores de la Tabla III, que se muestra a continuación.

Tabla III Valores de la Clasificación de imágenes

Valor	Descripción
0 a 100	Porcentaje de tonos de piel
-1	Archivo no encontrado
-2	Error de Licencia
-3	Formato de archivo erróneo
-4	Coincidencias no encontradas
-5	Folder o Directorio no encontrado
-6	Error Desconocido
-7	No se puede cargar la imagen (Imagen corrupta)
-8	Sin suficiente Información
-9	Sin perfil de detección de rostro
-10	No se puede abrir el directorio de detección de rostro
-11	Archivo de detección de rostro no encontrado

Tabla III (cont.) Valores de la Clasificación de imágenes

Valor	Descripción
-12	Clasificador de entrada no inicializado
-13	Fallo de perfil Init
-14	La ruta del archivo está vacía
-16	Los datos de la imagen están vacíos
-17	Control de coincidencias nulo
-18	Falta resultado de recuperación
-100	Formato de archivo no soportado
-101	Imagen en blanco y negro no soportada
-102	Imagen en escala de grises no soportada
-103	Imagen monocromática no soportada
-1000	Error Desconocido
-1001	El valor de la función EID produjo una excepción
-1002	El valor de la función EID produjo una excepción

2.3.2 Paraben's Porn Detection Stick

Paraben's Porn Detection Stick es un dispositivo USB que busca las imágenes existentes en una computadora, escaneándolas para encontrar contenido pornográfico y crea un reporte de las imágenes pornográficas detectadas como sospechosas. Incluso puede escanear imágenes borradas, así como las imágenes que están en la cache del explorador Mozilla Firefox.

El Porn Detection Stick ha implementado numerosos métodos y algoritmos para analizar cada una de las imágenes con la finalidad de determinar si la imagen analizada puede ser una imagen pornográfica [36]. Algunos de los métodos y algoritmos usados son los siguientes:

- Limpieza de la imagen (Image Cleanup): Dentro de la memoria virtual, el software hace una limpieza de la imagen para un óptimo análisis.
- Detección de Piel (Skin Detection): Identifica los tonos de piel y texturas dentro de la imagen.
- Eliminación del fondo (Background Elimination): Elimina los objetos del fondo de los objetos del primer plano, porque los objetos del fondo no altera los resultados.
- Detección de Bordos (Edge Detection): Identifica los bordes de los objetos.
- Separación de las partes del cuerpo (Body Part Separation): Separa las partes del cuerpo.
- Curvatura Negativa de Rechazo (Negative Curvature Rejection): Elimina los objetos con curvas negativas.
- Eliminación (Elimination): Elimina otros objetos.
- Detección de rostro: Identifica los rostros dentro de la imagen.

- Diseño de la distribución de las partes del cuerpo (Body Parts Layout Decision): Mapas de la distribución de las partes del cuerpo y diseño de la imagen.
- Decisión: Clasifica la imagen en base a los datos obtenidos.

El software funciona de una sencilla manera, para poder analizar las imágenes existentes en la computadora, o si se desea analizar un directorio en específico, se necesitan seguir los siguientes pasos:

1. Iniciar sesión como administrador en la computadora que se desea analizar.
2. Conectar el dispositivo USB en algún puerto USB.
3. Ejecutar el archivo PornDetectionStick.exe.
4. Configurar las opciones de búsqueda.
5. Seleccionar el directorio o unidad física a analizar, para inicializar la búsqueda.
6. Ver el resultado de la búsqueda, el cual es un directorio que se genera en la carpeta Reports, que contiene las imágenes encontradas como sospechosas.

La interfaz del usuario se muestra en la Figura 2.11, que es donde se selecciona la unidad física o directorio a analizar.

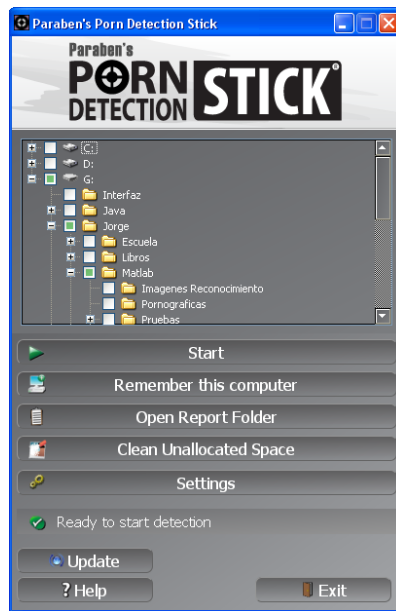


Fig.2.11 Interfaz Paraben's Porn Detection Stick

En la opción "Settings" se configura en una pantalla nueva las opciones de búsqueda, la pantalla se muestra en la figura 2.12.

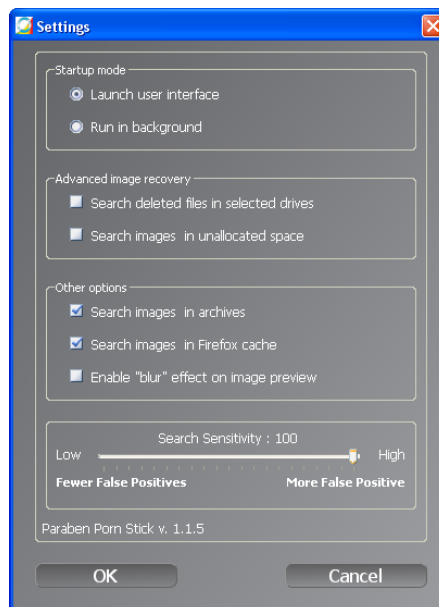


Fig.2.12 Configuración Paraben's Porn Detection Stick

La opción más importante a configurar es la de sensibilidad, ya que de ello depende en gran medida la cantidad de imágenes detectadas, porque es el umbral con el que las imágenes serán clasificadas, entre más alta sea la sensibilidad habrá mayor número de imágenes clasificadas como para adultos pero también existirán mayor número de falsos positivos dentro de las imágenes clasificadas como pornográficas.

El reporte de las imágenes detectadas como pornográficas se puede observar desde la misma interfaz como se muestra en la Figura 2.13, o accediendo al directorio donde se guardan los reportes creados.

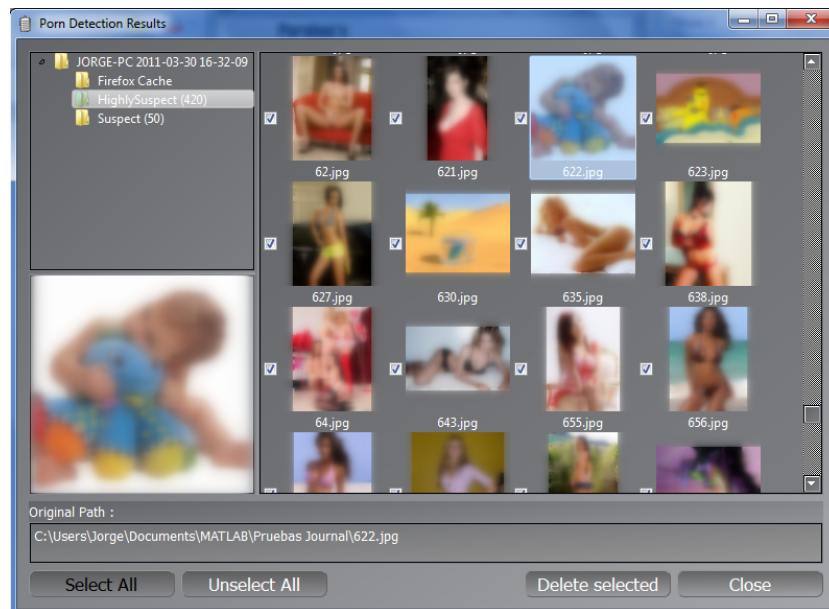


Fig.2.13 Reporte de imágenes detectadas

2.3.3 Otras Soluciones

Existen en el mercado algunas otras soluciones, que serán mencionadas de forma breve en esta sección, ya que sólo las dos anteriores pudieron ser adquiridas.

Entre las soluciones que se ofrecen son las siguientes:

a) Porn Terminator: Es un software que permite al usuario detectar las imágenes pornográficas existentes, así como los videos que pueda contener la computadora analizada.

b) Snitch de Hyperdyne Software: Este Software permite hacer una búsqueda en la computadora, para encontrar imágenes pornográficas.

c) SurfRecon Pornography Detection Software: Este software provee herramientas para detectar imágenes y videos del tipo pornográficos.

d) Porn-Detection Software for Videos & Images de YangSky: Este software permite detectar imágenes y videos clasificados como para adulto a través de una computadora.

CAPÍTULO 3: ANÁLISIS DE IMÁGENES EXPLÍCITAS

En el presente capítulo se describirá de manera detallada la metodología propuesta para poder clasificar imágenes en imágenes con contenido explícito o no. Además se describe el método utilizado para proponer un umbral ó el intervalo de valores de cada píxel para saber si éste puede ser tomado como piel, y de esta manera crear un filtro con la detección de piel.

3.1 Introducción

El desarrollo de una metodología propia para la detección de imágenes con contenido explícito o pornográfico se presenta en este capítulo.

El sistema propuesto que se ha desarrollado consta de dos módulos, el primero que se usó solamente para la búsqueda de un umbral en el que los píxeles pueden ser clasificados como piel o no, y el módulo de prueba, el cual tiene cuatro etapas para poder clasificar las imágenes de entrada.

Las etapas del sistema propuesto para el módulo de prueba son las siguientes:

- 1.- Transformación: La imagen es transformada al modelo de color deseado, en este caso puede ser al modelo de color HSV o YCbCr.
- 2.- Etapa de Filtrado: En esta etapa la imagen es filtrada usando el umbral que fue calculado en el módulo uno, para poder clasificar cada uno de los píxeles.
- 3.- Cuantificador: Una vez que la imagen ha sido filtrada, se cuenta la cantidad de píxeles detectados como piel.
- 4.- Decisión: Si la cantidad de píxeles detectados como piel son más de cierto porcentaje, la imagen será clasificada como con contenido explícito.

3.2 Diagrama del Sistema Propuesto

El sistema propuesto está dividido en dos módulos, donde cada módulo es representado en un diagrama de bloques para poder identificar de manera más sencilla los pasos que se siguen en cada uno de éstos.

3.2.1 Módulo del cálculo del umbral

Para poder determinar el umbral apropiado en cada uno de los modelos de color utilizado (HSV y YCbCr) fue necesario ocupar el diagrama a bloques que se muestra en la Figura 3.1, y así de esta manera proponer un umbral en el que los pixeles pueden ser clasificados como piel o no.

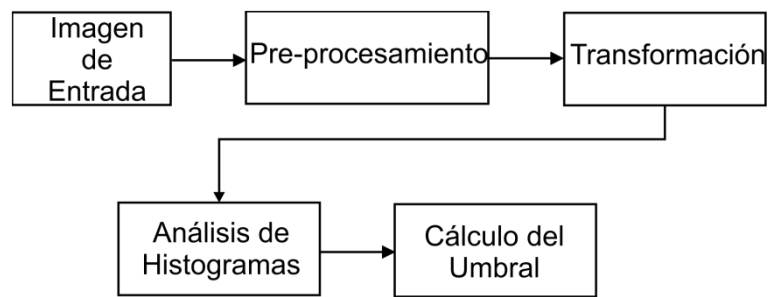


Fig. 3.1 Módulo Cálculo del Umbral

3.2.2 Módulo de Prueba

Una vez que el umbral para cada uno de los modelos de color fue calculado, se sigue la metodología mostrada en la Figura 3.2 para comprobar si una imagen es una imagen con contenido explícito o no.

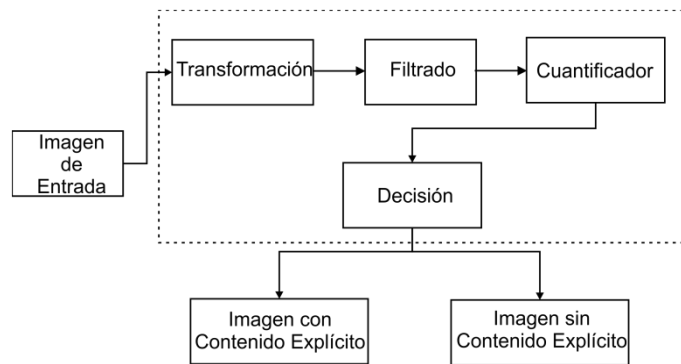


Fig. 3.2 Módulo de prueba del sistema propuesto

3.3 Metodología

Como se ha explicado anteriormente, el sistema propuesto consta de dos módulos, en este punto se explicará de manera detallada cada uno de las fases de éstos.

3.3.1 Módulo de Cálculo del umbral

Este módulo calcula un umbral de detección con el que los píxeles pueden ser clasificados como piel, y de esta manera descartar los que no son.

- **Pre-Procesamiento**

En esta etapa se usaron imágenes de personas de distintas razas para poder obtener los diferentes tonos de piel que existen en el mundo, las cuales fueron recortadas para obtener el rostro y de esta manera hacer un análisis de las zonas de piel que tiene el rostro.

Dentro de las imágenes existen personas como: Europeas, Latinas, Caucásicas, Asiáticas, Africanas. La figura 3.3 muestra una imagen original y una imagen recortada del rostro.

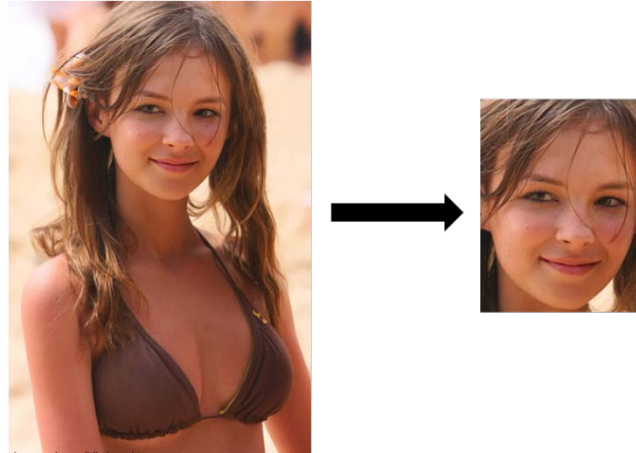


Fig. 3.3 Pre-procesamiento imagen

- **Transformación**

La imagen del rostro que fue recortada anteriormente es transformada al modelo de color HSV y YCbCr usando las ecuaciones (2) a (4) y (12), respectivamente.

La Figura 3.4 muestra la imagen original en el modelo RGB y la transformación al modelo HSV.

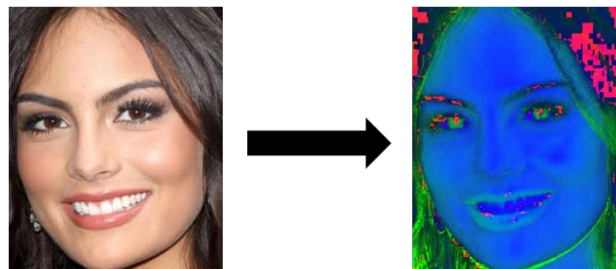


Fig. 3.4 Rostro de una persona con piel oscura

En la Figura 3.4 se puede observar que los tonos de piel se ven en un tono verde o azul, pero dependiendo de la iluminación con la que fue tomada la imagen, ésta cambiara y las zonas de piel se verán afectadas.

La Figura 3.5 muestra el ejemplo de la transformación al modelo YCbCr de una imagen de un rostro de una persona con piel blanca.

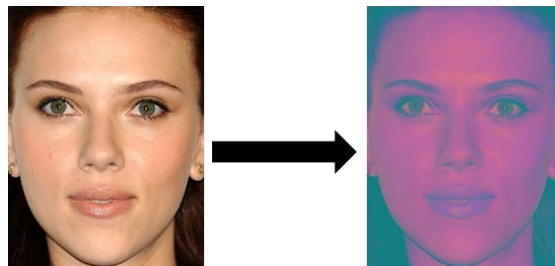


Fig. 3.5 Rostro de una persona con piel blanca

En la Figura 3.5 se observa que las zonas de piel son de algún tono como morado, esto es por la combinación de las componentes de crominancia roja y azul de este modelo de color.

- **Análisis de Histogramas en el modelo *HSV***

Una vez hecha la transformación de la imagen, la imagen es separada en cada una de sus componentes y así de esta manera observar cuales son los valores de los píxeles que tienen algún tono de piel. La Figura 3.6 se muestra un ejemplo de las componentes de una imagen en formato HSV.

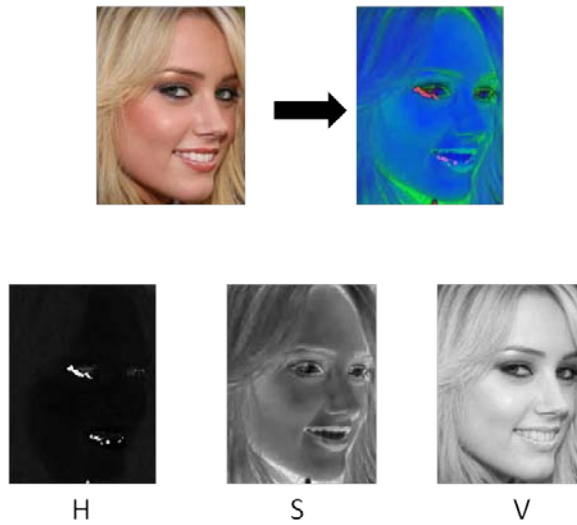


Fig. 3.6 Componentes de imagen *HSV*

Para determinar el umbral para clasificar los píxeles como piel, se analiza el histograma de cada uno de las componentes en el modelo HSV de diferentes tipos de piel para poder abarcar al máximo todas los tipos de piel existentes en las imágenes en Internet.

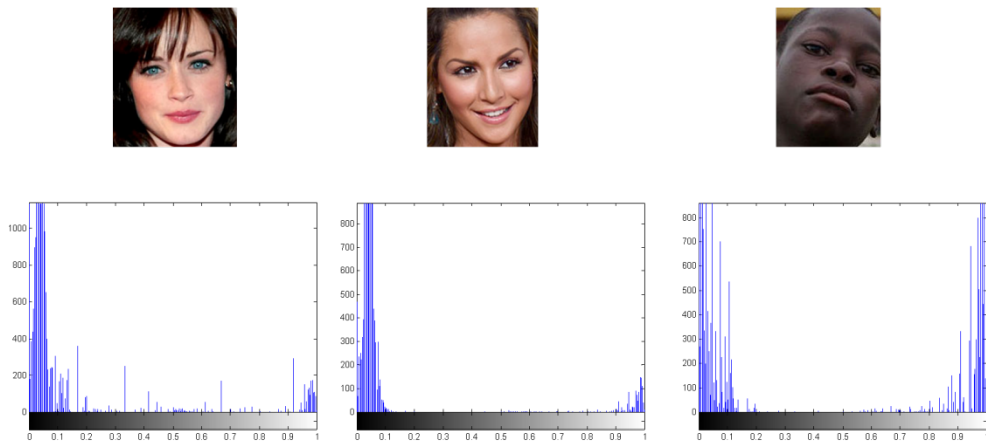


Fig. 3.7 Histogramas de la componente *H*

En la Figura 3.7 se puede observar que el valor máximo en los distintos tonos de piel es **0.12** para la componente H, por lo que el umbral propuesto en el modelo *HSV* tendrá como intervalo para ésta componente los siguientes valores: **H= [0, 0.12]**.

Del mismo modo se hizo el análisis de la componente S, obteniendo como resultado el umbral de detección de **S= (0.15, 0.9)**, en la Figura 3.8 se muestran sólo algunos ejemplos de los histogramas utilizados para el análisis.

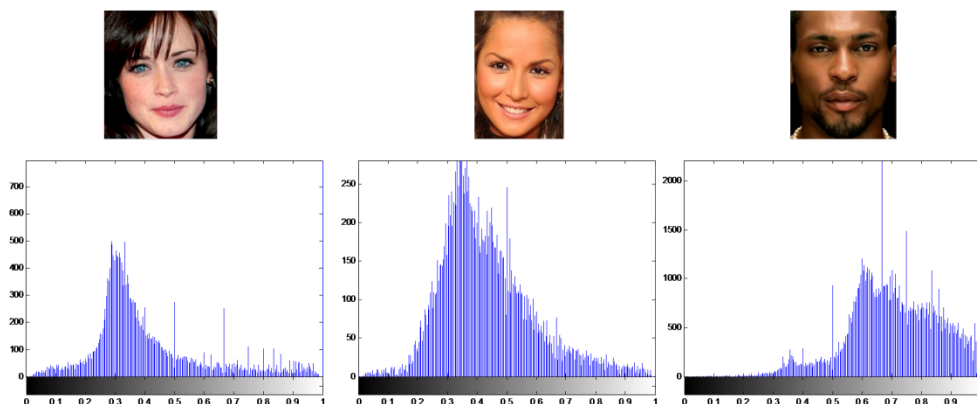


Fig. 3.8 Histogramas de la componente S

El análisis del histograma de la componente V dio como resultado el siguiente umbral, **V= (0.15, 0.9)** algunas imágenes y sus respectivos histogramas de la componente V son mostrados en la Figure 3.9.

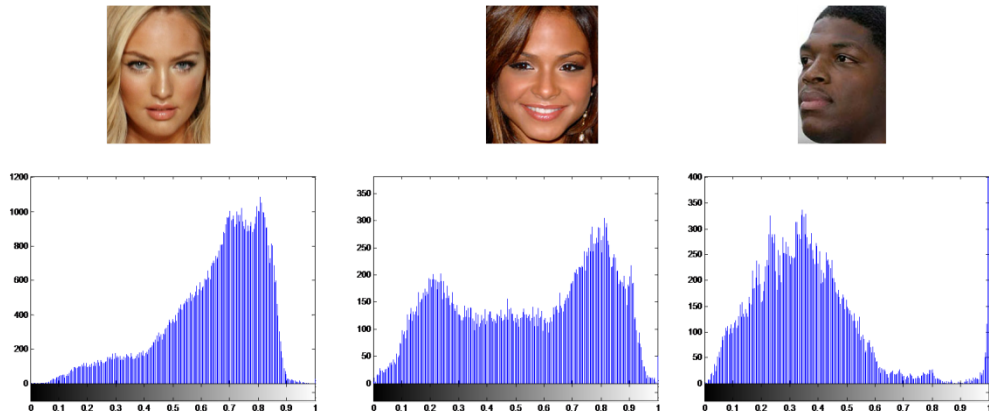


Fig. 3.9 Histogramas de la componente V

Resumiendo lo anterior de cada uno de los intervalos para cada componente se puede decir que las reglas de decisión para la clasificación de píxeles con algún tono de piel es el que se muestra en (24).

$$\begin{aligned}
 0 < H &\leq 0.12 \text{ y} \\
 0.15 < S &< 0.9 \text{ y} \\
 0.2 < V &< 0.95
 \end{aligned}
 \tag{24}$$

Usando (24), se realiza la detección de piel como se muestra en la Figura 3.10, la cual es un ejemplo de esta detección.

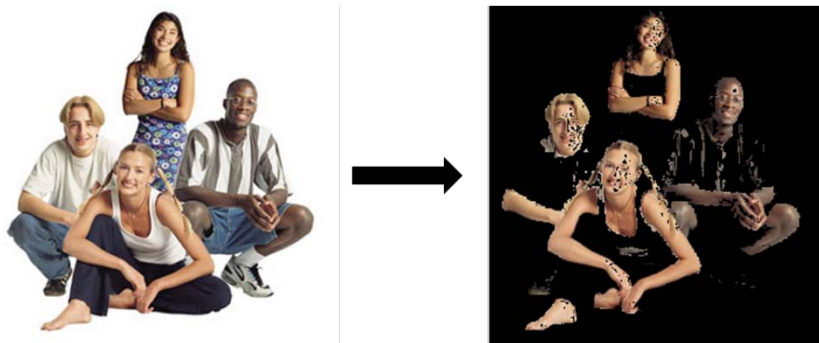


Fig.3. 10 Detección de Piel usando el umbral del modelo HSV

- **Análisis de Histogramas en el modelo *YCbCr***

El paso siguiente después de haber realizado la transformación del modelo RGB a YCbCr es separar cada una de las componentes de la imagen transformada para poder analizar sus histogramas, y de esta manera proponer un intervalo en el que los píxeles pueden ser clasificados como piel o no. Un ejemplo de una imagen transformada y sus componentes son mostradas en la Figura 3.11.

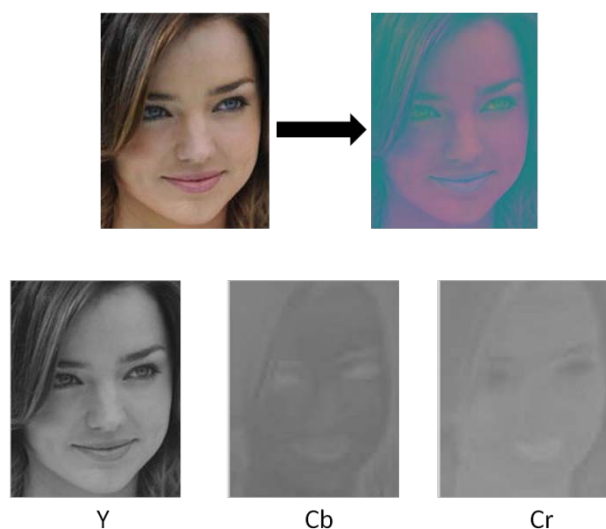


Fig. 3.11 Imagen Transformada y sus componentes Modelo YCbCr

En este modelo sólo se tomo las componentes Cb y Cr debido a que los histogramas de la componente Y para los diferentes tipos de piel son muy distintos y es muy difícil proponer un intervalo en esta componente por dicha desigualdad. En la Figura 3.12 se podrá observar con mayor claridad las diferencias de algunos histogramas con distintos tonos de piel.

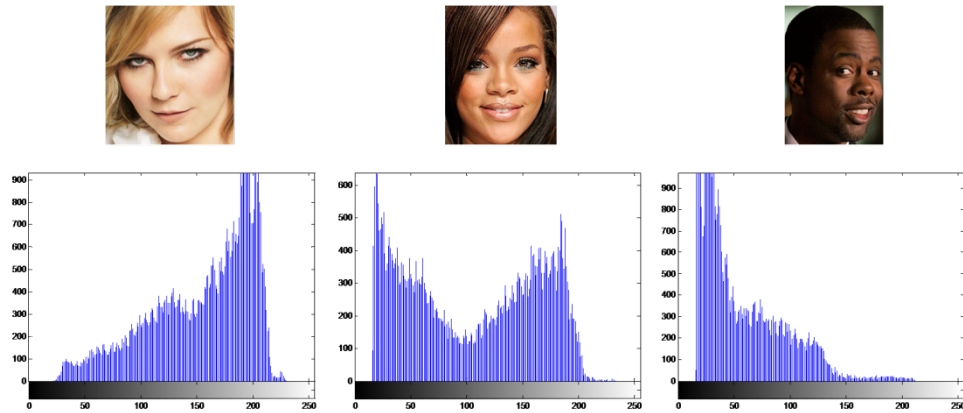


Fig. 3.12 Histogramas de la componente Y

Los histogramas de la componente Cb de algunas imágenes del rostro con distinto tono de piel son mostrados en la Figura 3.13.

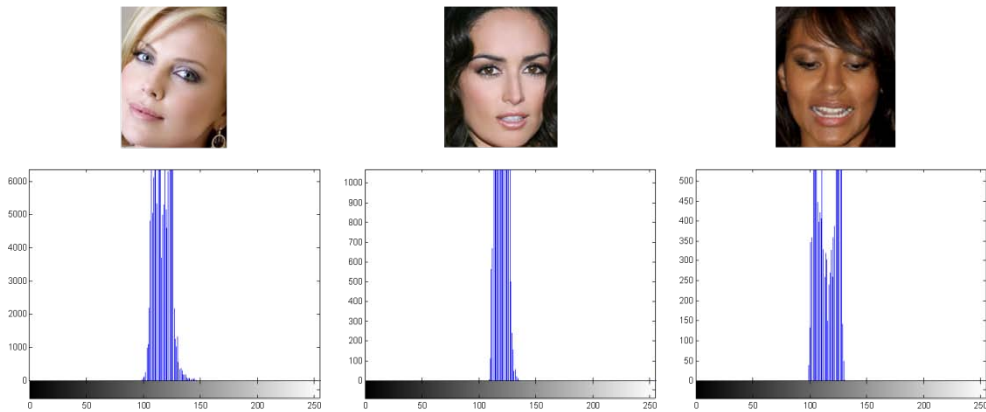


Fig. 3.13 Histogramas de la componente Cb

En la Figura 3.13 se puede observar que el intervalo existente entre los diferentes tonos de piel está entre el rango de valores de **80** y **120**, por lo que este rango será nuestro umbral de ésta componente para poder descartar los píxeles que no son piel.

Del mismo modo se hizo un análisis de los histogramas de distintas imágenes de la componente Cr para poder determinar un umbral que permita clasificar los píxeles como piel. En la Figura 3.14 se muestra un ejemplo de algunos rostros utilizados durante el análisis de histogramas

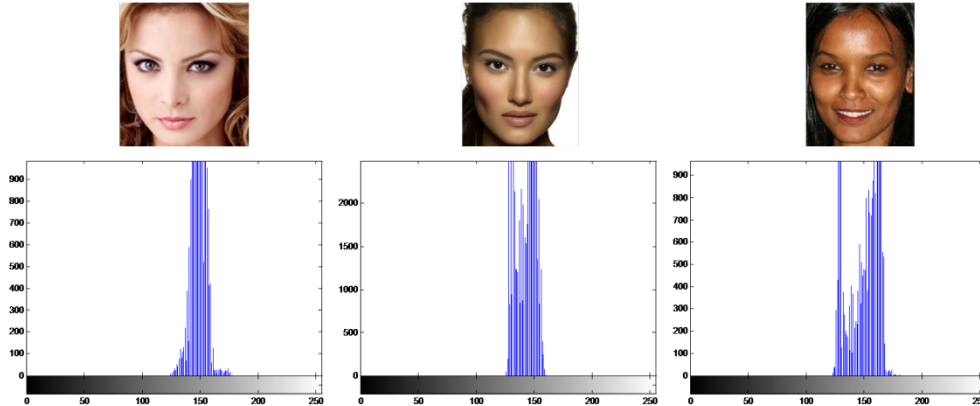


Fig. 3.14 Histogramas de la componente Cr

Para el caso de la componente Cr se pudo determinar que el umbral donde se pueden clasificar los píxeles está entre el rango de valores de **133** y **173**, esto debido a las variaciones de los tonos de piel, los cuales van desde una piel blanca a una piel de color negro.

Recopilando la información anterior se puede decir que el umbral determinado para este modelo es el que se muestra en (25), el cual se debe de cumplir para decretar que el píxel es un píxel con algún tono de piel.

$$80 \leq Cb \leq 120 \quad y \quad 133 \leq Cr \leq 173 \quad (25)$$

Para observar el funcionamiento del umbral propuesto en la Figura 3.15 es aplicada la regla anterior en una imagen con distintas personas, al final sólo se pueden observar los píxeles detectados como piel, los demás píxeles son convertidos en negro, esto con la finalidad de que los píxeles con algún tono de piel humana puedan ser vistos de mejor manera.

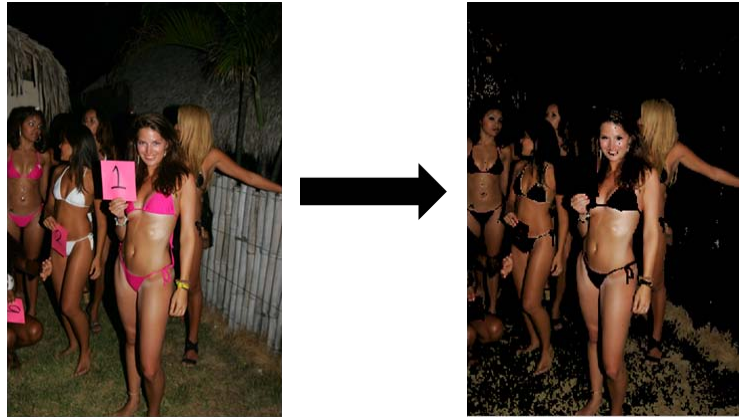


Fig. 3.15 Detección de Piel con Umbral del modelo YCbCr

3.3.2 Módulo de Prueba

Este módulo es utilizado para analizar las imágenes y poder determinar si éstas son imágenes con contenido explícito o no, cada una de las etapas de este modulo son descritas a continuación.

- **Transformación**

La primera etapa del módulo de prueba es convertir la imagen de entrada, que normalmente está en el modelo RGB al modelo HSV o YCbCr usando las expresiones matemáticas (2) a (4) para el modelo HSV y la expresión (12) para el otro modelo.

La selección de uno de los modelos anteriormente mencionados dependerá del usuario, la cual será una opción en una interfaz gráfica que se verá más adelante. Además se hizo para saber qué modelo es más eficaz para este tipo de tarea.

- **HSV**

Las Figuras 3.16 y 3.17 muestran una imagen de un paisaje y de una persona, respectivamente, éstas serán utilizadas como ejemplo para observar el desarrollo del sistema en el modelo HSV.

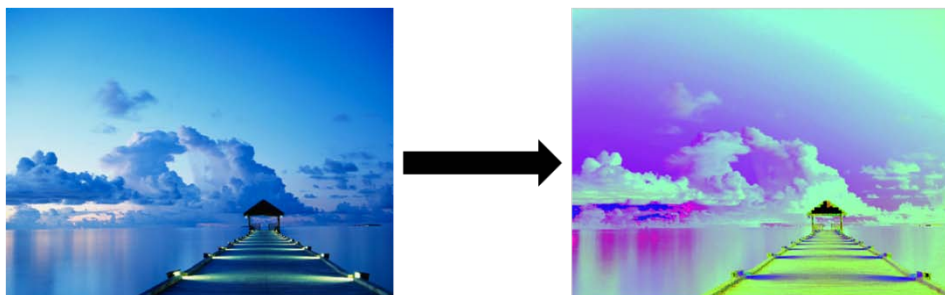


Fig. 3.16 Transformación de una imagen de un paisaje

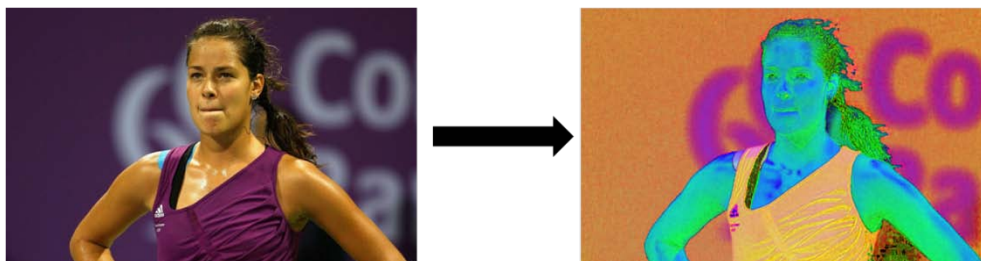


Fig. 3. 17 Transformación de una imagen de una persona

- **YCbCr**

Para este modelo también se tomarán dos ejemplos para explicar el funcionamiento del sistema, una imagen sin personas y otra donde aparezcan personas las cuales son mostradas en la Figura 3.18 y 3.19, respectivamente.

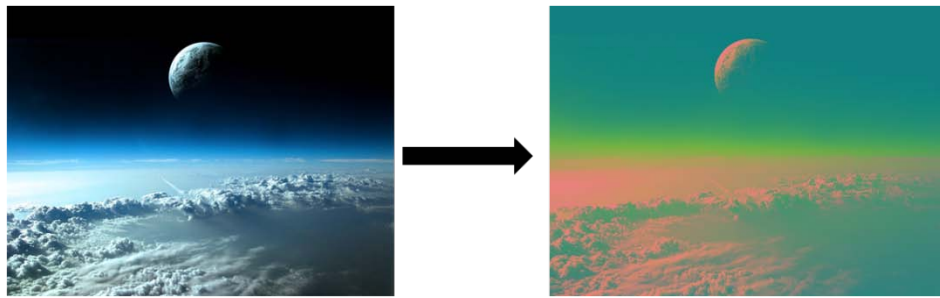


Fig. 3.18 Transformación de una imagen

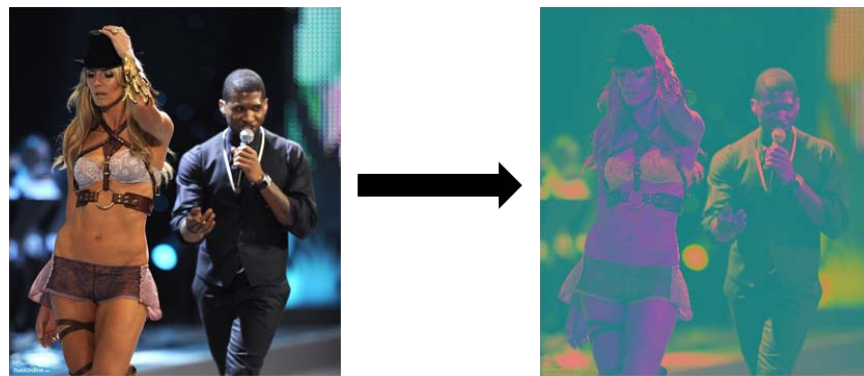


Fig. 3.19 Transformación de una imagen de personas

- **Filtrado**

Una vez hecha la transformación de la imagen, la segunda etapa del sistema es filtrar la imagen usando la detección de piel, para lograr esto se ocupan los umbrales definidos en (24) y (25). La finalidad de la detección de piel es poder discriminar todos aquellos píxeles que no contengan algún tono de piel.

- **HSV**

Las imágenes mostradas en las Figuras 3.16 y 3.17 son filtradas usando (24) para obtener los píxeles que tengan algún tono de piel, el resultado de esta operación es mostrada en las Figuras 3.20 y 3.21.

Para obtener la imagen filtrada, primeramente la imagen transformada es revisada píxel por píxel y si la regla del umbral se cumple, el valor del píxel de la imagen original es asignado a la imagen filtrada, si el píxel no cumple con la regla el valor asignado a la imagen filtrada es de cero, esto con la finalidad de poner de color negro lo que no sea reconocido como piel.

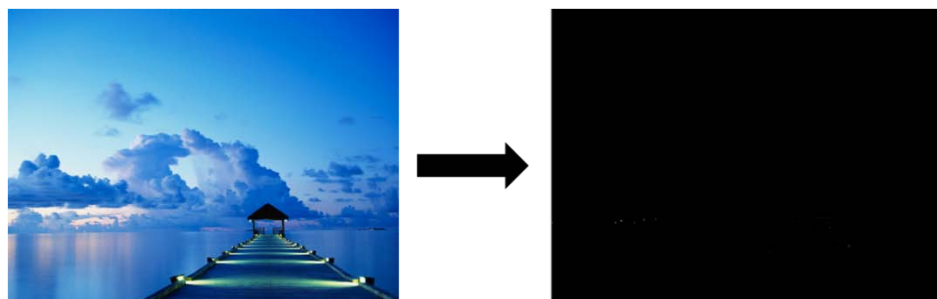


Fig. 3.20 Imagen de un paisaje filtrada

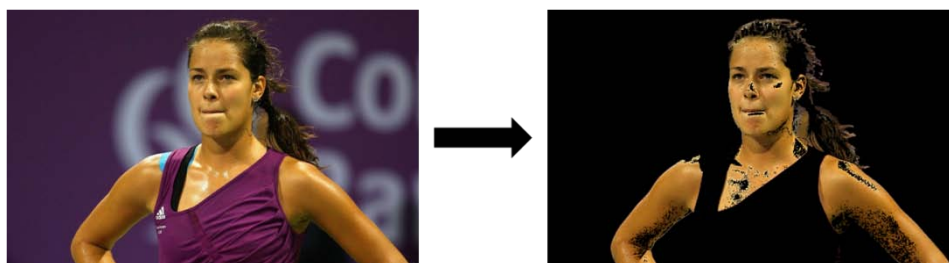


Fig. 3.21 Imagen filtrada de una persona

○ **YCbCr**

Las imágenes mostradas en las Figuras 3.18 y 3.19 son filtradas usando la detección de piel, que se hace utilizando la regla (25), las imágenes resultantes son mostradas en las Figuras 3.22 y 3.23.

La imagen filtrada se obtiene analizando la imagen transformada píxel por píxel, si la regla del umbral es cumplida el valor de la imagen original es asignado a la imagen filtrada, en caso contrario el valor asignado es cero.

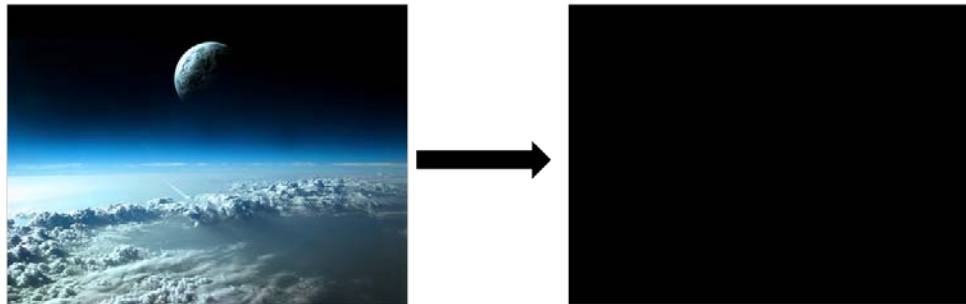


Fig. 3.22 Imagen Filtrada

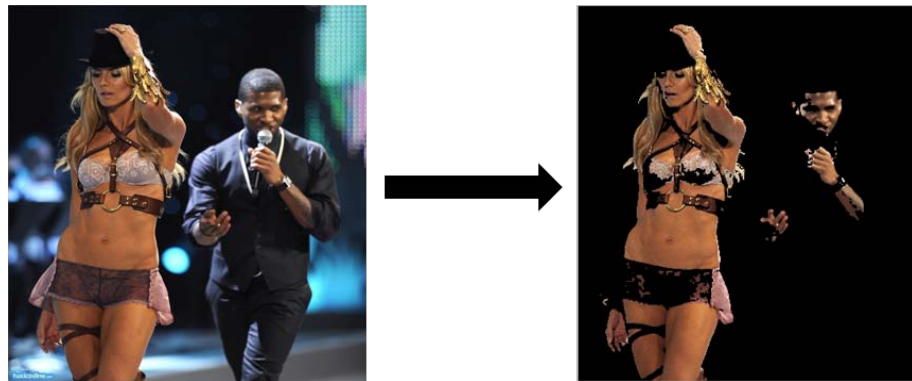


Fig. 3.23 Imagen Filtrada de dos personas

- **Cuantificador**

Al finalizar la etapa de filtrado, se tiene una imagen con sólo las zonas de piel que fueron detectadas como tal, para conocer la cantidad de piel que fue detectada se cuentan todos los píxeles que sean diferentes de cero o diferentes de negro, después de saber el número de píxeles detectados se ocupa la expresión mostrada en (26) para determinar el porcentaje piel existente en la imagen.



$$\text{porcentaje de piel} = \frac{\# \text{ píxeles de la zona más grande de piel}}{\# \text{ total de píxeles de color de piel}} \times 100 \quad (26)$$

En la Tabla IV se muestran los porcentajes de las cuatro imágenes que se mostraron anteriormente en esta sección.

Tabla IV Porcentaje de Piel de Imágenes

Imagen de entrada	Modelo de color	Imagen Filtrada	Porcentaje de Piel
	HSV		0%
	HSV		48.84%
	YCbCr		0%

Tabla IV (cont.) Porcentaje de Piel de Imágenes

	<p>YCbCr</p>		<p>82.75%</p>
---	--------------	--	---------------




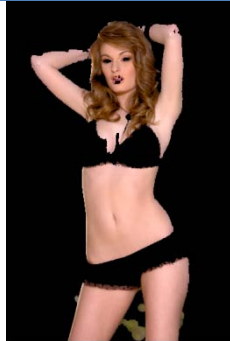
- **Decisión**

Al finalizar la etapa de cuantificación se toma una decisión de acuerdo al porcentaje de piel que fue detectado, si la imagen contiene más del **58%** de piel, ésta se clasificará como una imagen con contenido explícito, de otra manera la imagen será clasificada como imagen sin contenido explícito.

La decisión de tomar ese porcentaje de piel, es debido a que las imágenes pornográficas o con contenido explícito son en mayor parte piel, por lo que estas zonas ocupan cerca de la mitad de la imagen.

En la Tabla V se muestran algunos ejemplos de la clasificación de imágenes con contenido explícito usando la metodología propuesta en esta sección.

Tabla V Clasificación de imágenes

Imagen de entrada	Modelo de color	Imagen Filtrada	Porcentaje de Piel
	HSV		99.26%
	HSV		49.93%
	YCbCr		84.28%
	YCbCr		46.08%

3.3.3 Diseño de la Interfaz

La interfaz desarrollada consta de dos módulos para el análisis de imágenes, el primero que solamente analiza una imagen, y el segundo que analiza el contenido de una carpeta o folder dentro de la computadora.

La interfaz se muestra en la Figura 3.24, esta es la ventana principal donde se seleccionara dependiendo de lo que se desee analizar.

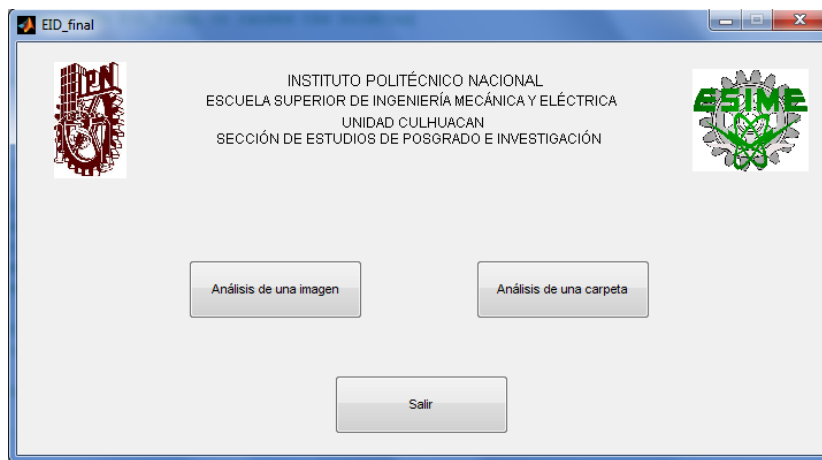


Fig. 3.24 Interfaz Desarrollada

- **Módulo de análisis de una imagen**

En este módulo, se desarrollo una interfaz capaz de analizar cualquier tipo de imagen sin importar el tipo de archivo, siempre y cuando el tipo de archivo sea soportado por MATLAB y la imagen sea a color. Los tipos de archivos soportados por MATLAB son los siguientes: *.bmp, *.jpg, *.png, *.gif, *.ppm, entre otros.

Lo primero que se debe de hacer para analizar una imagen, es seleccionar la imagen deseada, el segundo paso es elegir el modelo de color

con el que se desea examinar a ésta. En la Figura 3.25 se muestra la interfaz desarrollada para este módulo.

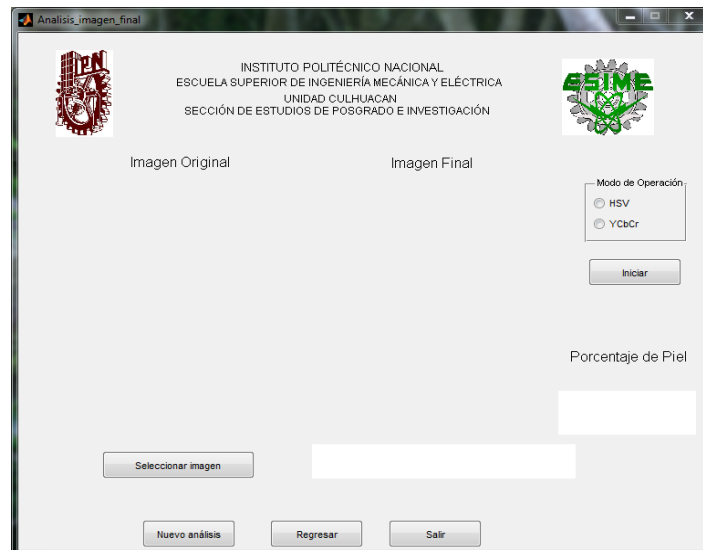


Fig. 3.25 Interfaz para análisis de una imagen

En las Figuras 3.26 y 3.27 se muestran ejemplos de una imagen sin contenido explícito y con contenido explícito, respectivamente.

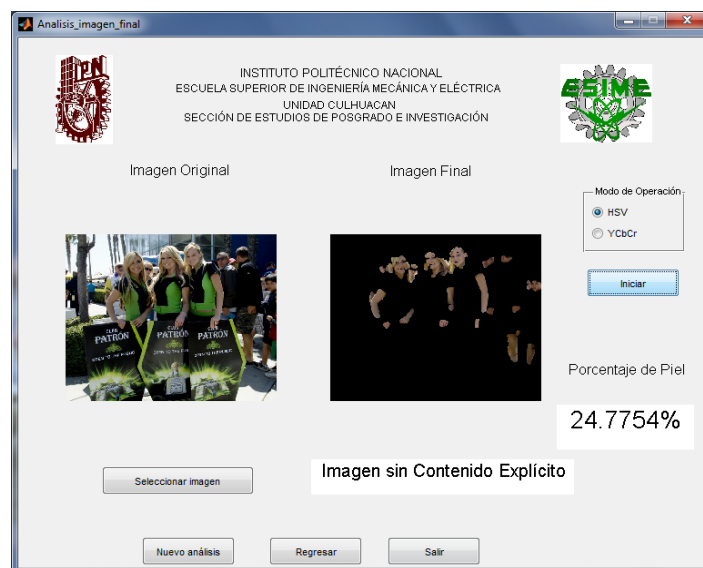


Fig. 3.26 Análisis de una imagen sin contenido Explícito

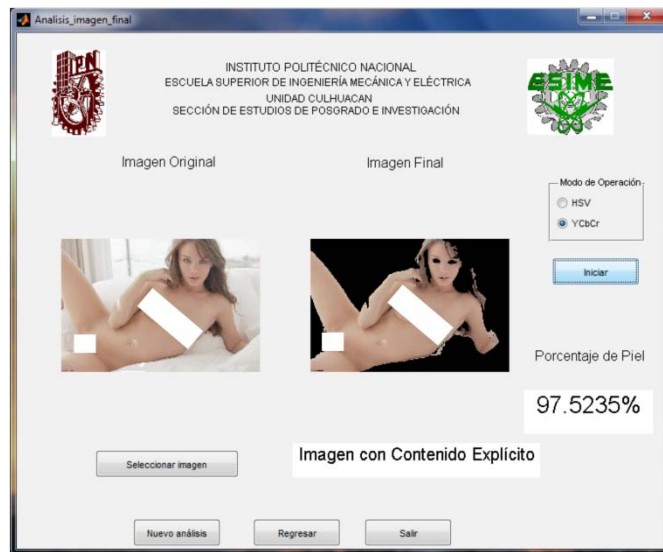


Fig. 3.27 Análisis de una imagen con contenido Explícito

- **Módulo de análisis de una carpeta**

En este modo el programa busca las imágenes contenidas dentro de un directorio en específico, para poder determinar de ese conjunto cuales son las imágenes que tienen contenido explícito, es una interfaz sencilla de utilizar, ya que sólo se debe de seleccionar algunos parámetros para poder iniciar la búsqueda. La interfaz diseñada para este módulo se muestra en la Figura 3.28. Los parámetros a seleccionar son:

- Buscar Directorio: Se elige el directorio a analizar.
- Guardar en: Se selecciona la ruta o dirección donde se guardará el folder con las imágenes detectadas como con contenido explícito.
- Modo de operación: Se escoge con cuál de los dos modelos de color se desea analizar.
- Tipos de archivos: Se seleccionan los tipos de imágenes a analizar (*.jpg, *.bmp, *.png).
- Nueva búsqueda: Limpia las variables, así como las opciones, para realizar una búsqueda en otro directorio

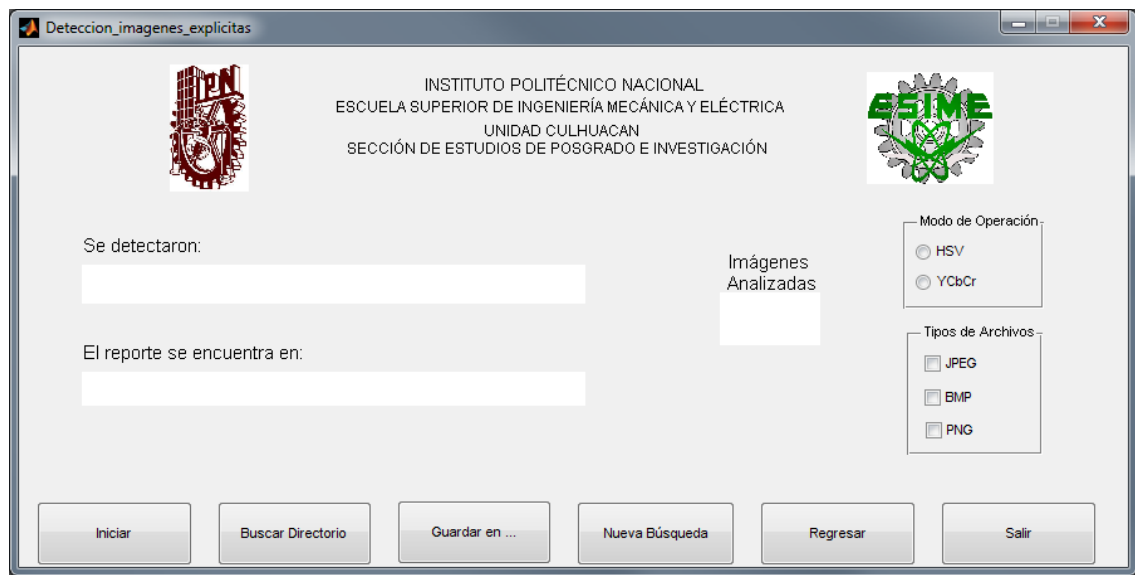


Fig. 3.28 Interfaz para analizar un directorio

En la Figura 3.29 se muestra un ejemplo del análisis de un directorio utilizando la interfaz mencionada anteriormente.



Fig. 3.29 Análisis de un directorio

CAPÍTULO 4: RESULTADOS EXPERIMENTALES

En este capítulo se presentan los resultados obtenidos usando tres conjuntos de imágenes distintos, usando el sistema propuesto, el software FTK 3.1 y el software Paraben's Porn Detection Stick.

4.1 Introducción

Los resultados obtenidos están divididos en secciones diferentes, en la primera sección se encuentran los resultados derivados del sistema propuesto usando los dos modelos de color ocupados a lo largo de esta tesis, en la segunda sección están aquellos resultados usando el módulo Explicit Image Detection del software FTK 3.1, y por último se detallan los resultados encontrados con el software Paraben's Porn Detection Stick.

Los tres sistemas se sometieron a distintas pruebas en las cuales se usaron distintos conjuntos de imágenes, el primero contenía 1200 imágenes con contenido explícito, el segundo conjunto contenía 1200 imágenes naturales¹, y el último conjunto de 2400 imágenes con 1200 imágenes con contenido explícito y 1200 imágenes naturales.

4.2 Sistema Propuesto

Para realizar las pruebas se utilizó la interfaz diseñada, la cual permite el análisis del contenido de un directorio en específico.

4.2.1 Modelo HSV

El primer conjunto de imágenes fue analizado usando este modelo, y se pudieron reconocer 996 imágenes como imágenes con contenido explícito, el resultado de éste análisis es mostrado en la Figura 4.1.

¹ Imagen Natural: Es aquella que por su naturaleza no contiene contenido explícito o pornográfico, por ejemplo: Personas vestidas, animales, plantas, automóviles, dibujos animados, paisajes, etc.

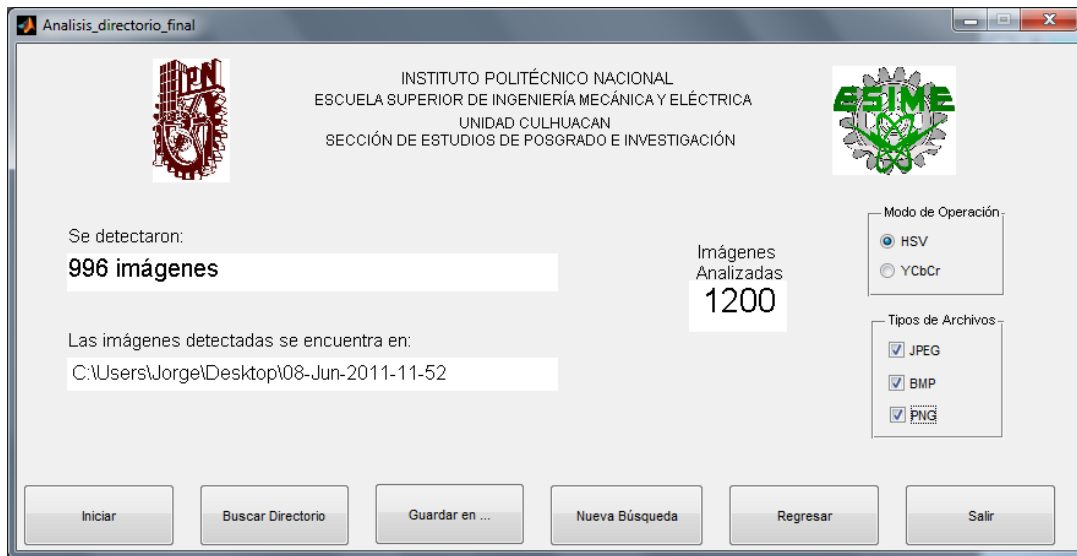


Fig. 4.1 Resultados de la primera prueba

Para el segundo caso se reconocieron 354 imágenes como imágenes clasificadas como para adultos del número total del conjunto, en la Figura 4.2 se puede observar el resultado que da la interfaz.

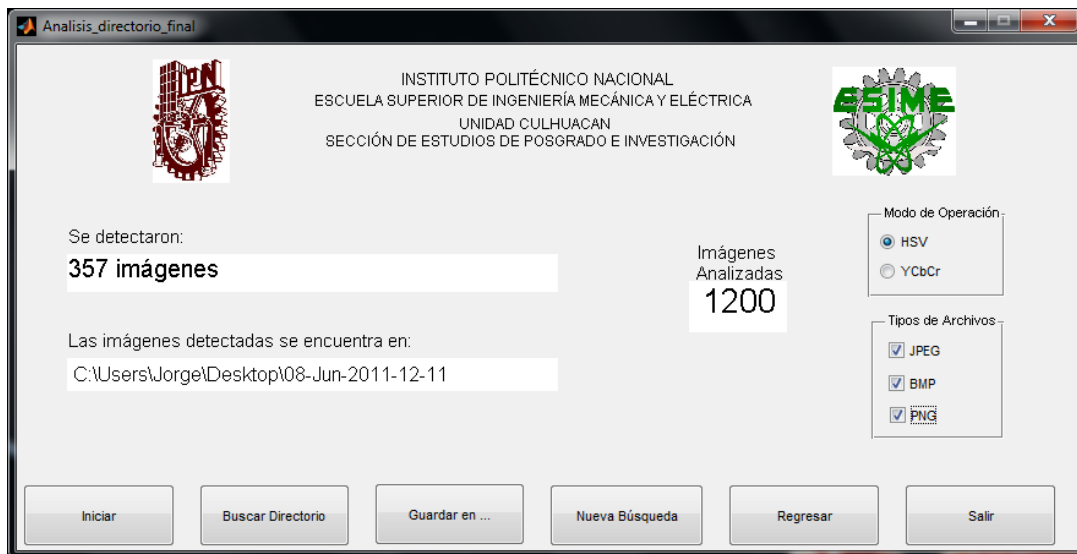


Fig. 4.2 Resultados del Segundo Conjunto de Imágenes

Por último se probó al sistema con el tercer conjunto de imágenes, del cual se reconocieron 1353 imágenes como imágenes con contenido explícito, en la Figura 4.3 se puede observar el resultado obtenido de la interfaz.



Fig. 4.3 Resultados de la última prueba

En la Figura 4.4 se muestra una gráfica donde se puede observar de manera el porcentaje de reconocimiento, así como el porcentaje de falsos positivos que se obtuvo al ser probado el sistema propuesto con tres diferentes conjuntos de imágenes.

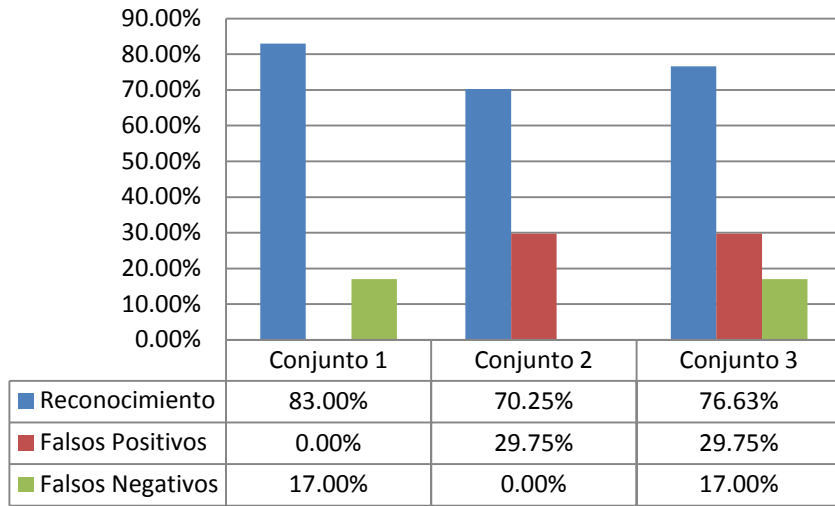


Fig. 4.4 Porcentajes de Reconocimiento Sistema Propuesto HSV

Analizando los porcentajes obtenidos se puede decir que usando el rango de valores para realizar la detección de piel en el modelo de color HSV se puede obtener un 83.00% de reconocimiento de imágenes con contenido explícito, con un 29.75% de falsos positivos y 17% de Falsos Negativos.

4.2.2 Modelo YCbCr

En la primera prueba el sistema reconoció 1060 imágenes como imágenes con contenido explícito, este resultado se puede observar en la Figura 4.5.



Fig. 4.5 Resultado de Reconocimiento del Conjunto 1 con YCbCr

En el segundo experimento se detectaron 443 imágenes como con contenido explícito del conjunto de 1200, la Figura 4.6 muestra el resultado de la detección usando la interfaz.



Fig. 4.6 Resultado de detección del segundo conjunto

Por último con el tercer conjunto el sistema identifico 1817 como imágenes con contenido explícito, dentro de estas imágenes reconocidas existen algunas que no lo son. La interfaz diseñada para hacer este reconocimiento es mostrada en la Figura 4.7 con el resultado de esta prueba.



Fig. 4.7 Resultado del tercer conjunto de pruebas

En la Figura 4.8 se muestra una gráfica donde se puede la tasa de reconocimiento, así como el porcentaje de falsos positivos que se obtuvo durante las pruebas al sistema usando tres diferentes conjuntos de imágenes.

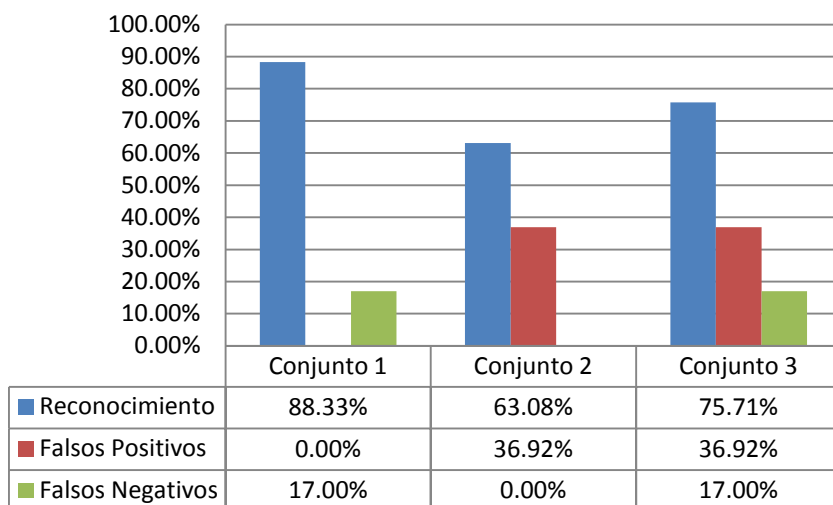


Fig. 4.8 Porcentajes de Reconocimiento, Falsos Positivos y Falsos Negativos con Modelo de Color YCbCr

Analizando los porcentajes se puede concluir que usando el modelo de color YCbCr como detector de piel, debido a el intervalo de valores encontrados para tal labor anteriormente, se puede obtener un 88.30% de reconocimiento de las imágenes, con un 36.92% de falsos positivos y 17% de Falsos Negativos.

4.3 FTK 3.1 Explicit Image Detection

Todas las pruebas realizadas se hicieron usando evidencia en vivo, la cual se obtiene en el momento de iniciar el análisis, en la Figura 4.9 se muestra la ventana de adquisición de la evidencia.

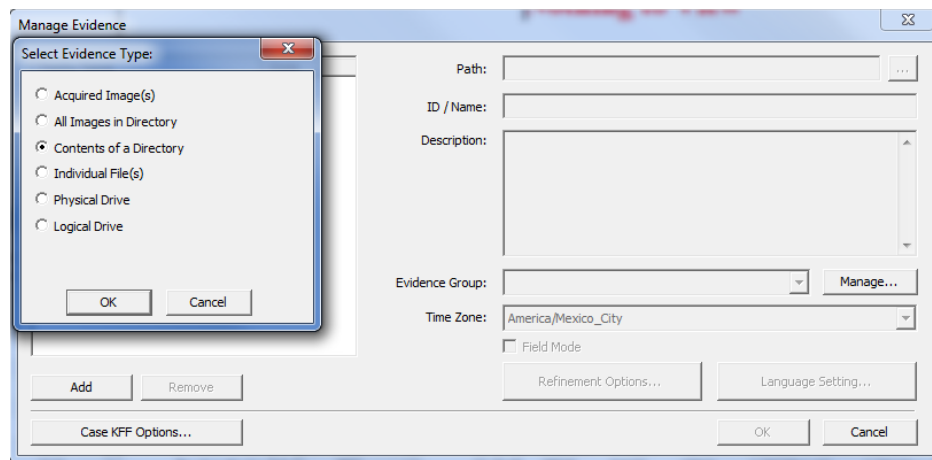


Fig. 4.9 Ventana de adquisición de la evidencia

Al realizar el análisis sólo se puede evaluar a las imágenes usando dos de los algoritmos que tiene el software, como se puede observar en la Figura 4.10.

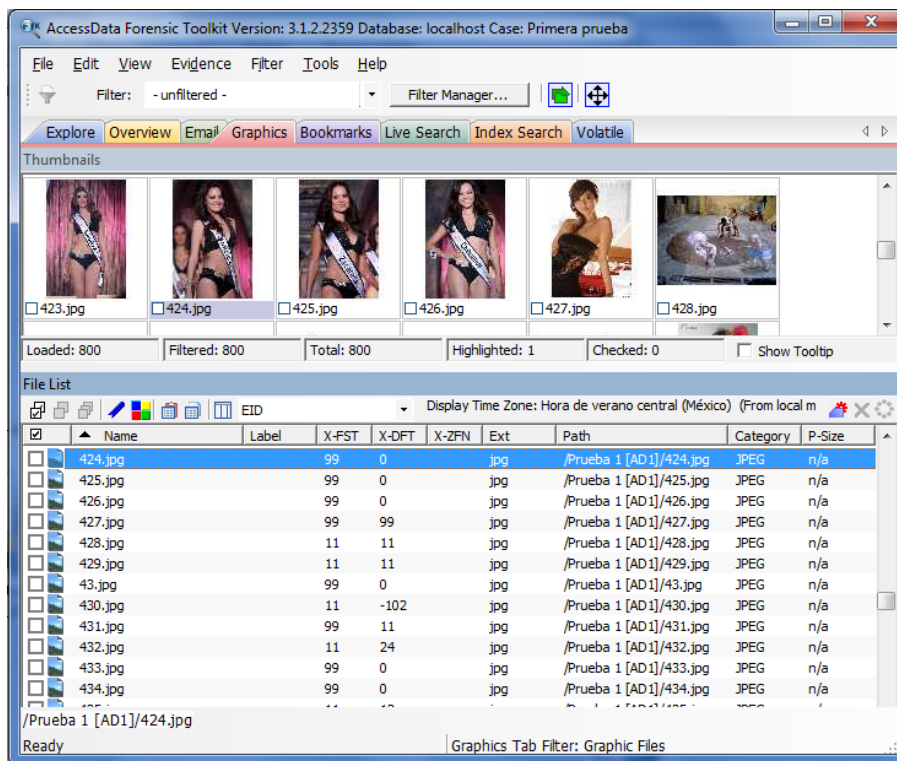


Fig. 4. 10 Primer análisis usando FTK 3.1

Para poder obtener el resultado de la evaluación del algoritmo que no fue utilizado en la primera evaluación, se ejecuta un análisis adicional que se puede seleccionar en la pestaña “Evidence”, en la Figura 4.11 se observa la ventana del análisis adicional.

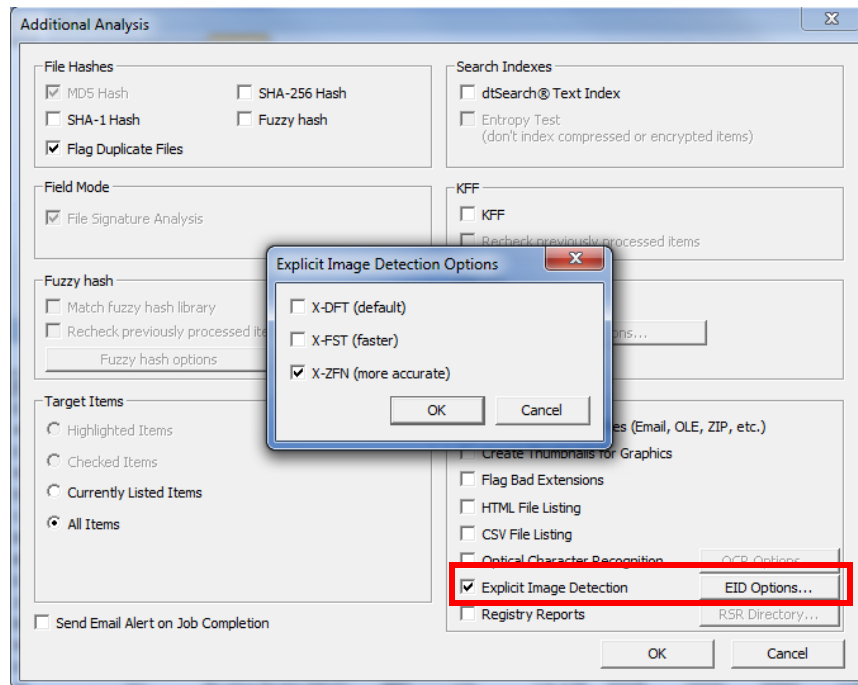


Fig. 4.11 Ventana del análisis adicional

Una vez hecho el segundo análisis en la ventana principal se pueden observar los valores que cada algoritmo le dio a cada una de las imágenes examinadas, como se puede observar en la Figura 4.12 que se muestra a continuación.

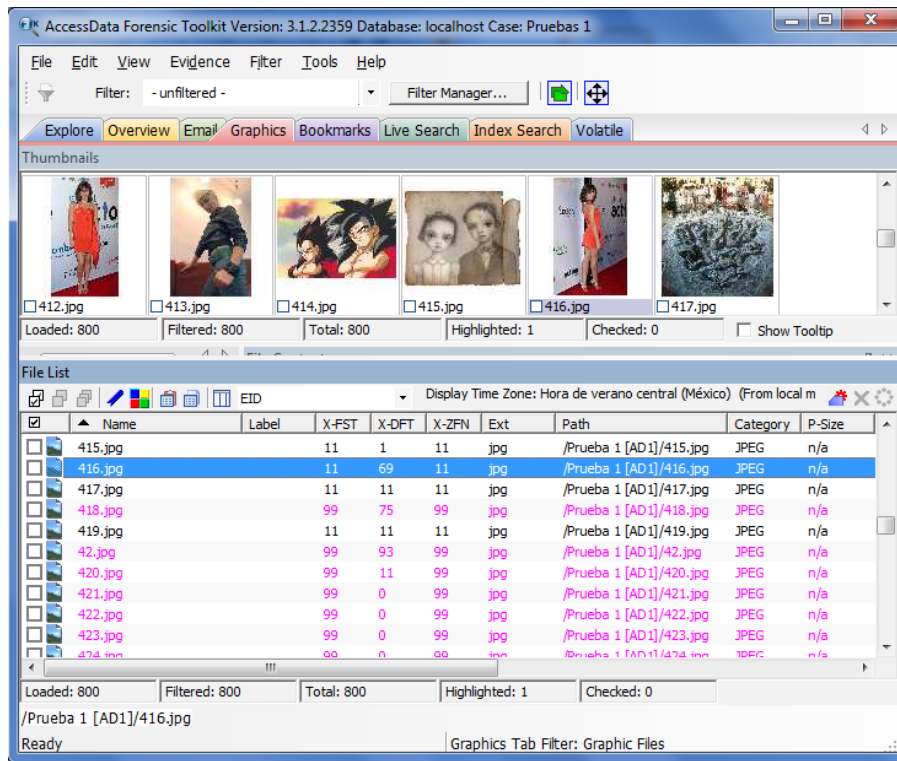


Fig. 4.12 Análisis de las imágenes

Existe una desventaja en el software ya que las imágenes deben ser clasificadas manualmente, las cuales dependerán de la persona que haga el análisis, los algoritmos sólo proporcionan valores de la cantidad de tonos de piel existentes en cada una de las imágenes.

Para realizar una clasificación más rápida se generan filtros para poder obtener sólo las imágenes que al menos dos de los algoritmos hayan dado un valor mayor a 80, esto debido a que la mayoría de las imágenes explícitas están en el rango de 80 a 100. En las Figuras 4.13, 4.14 y 4.15 se muestran los filtros generados.

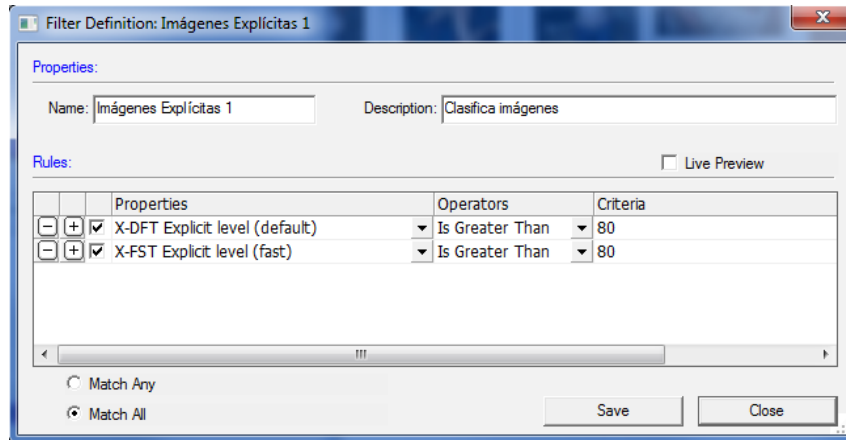


Fig. 4.13 Filtro para el primer y segundo algoritmo

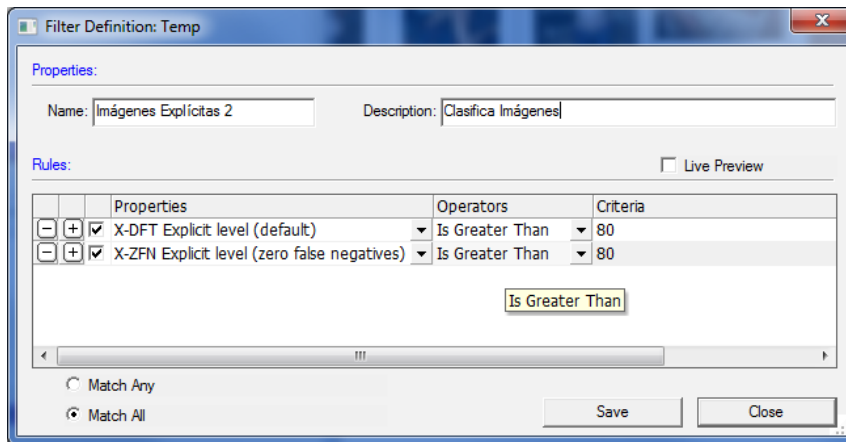


Fig. 4.14 Filtro para el segundo y tercer algoritmo

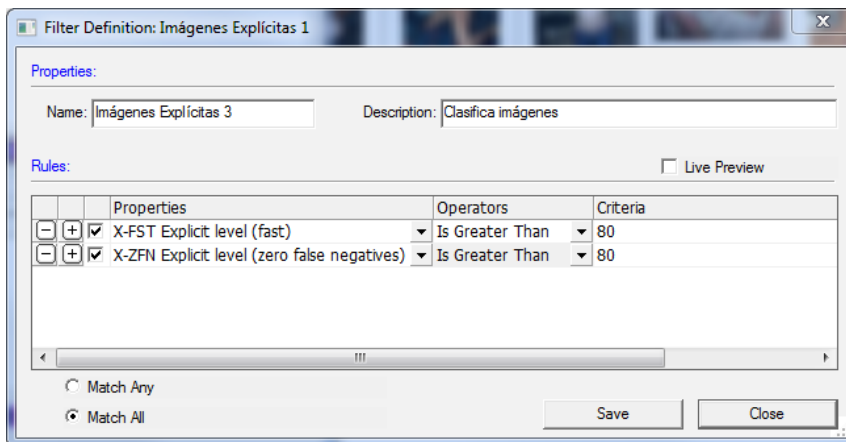


Fig. 4.15 Filtro para el primer y tercer algoritmo

Una vez creados los filtros se necesita generar una regla para que sean tomados en cuenta los filtros creados anteriormente, en la Figura 4.16 se observa que cualquier imagen que al menos dos de los algoritmos haya dado un valor mayor a 80 será clasificada como imagen con contenido explícito.

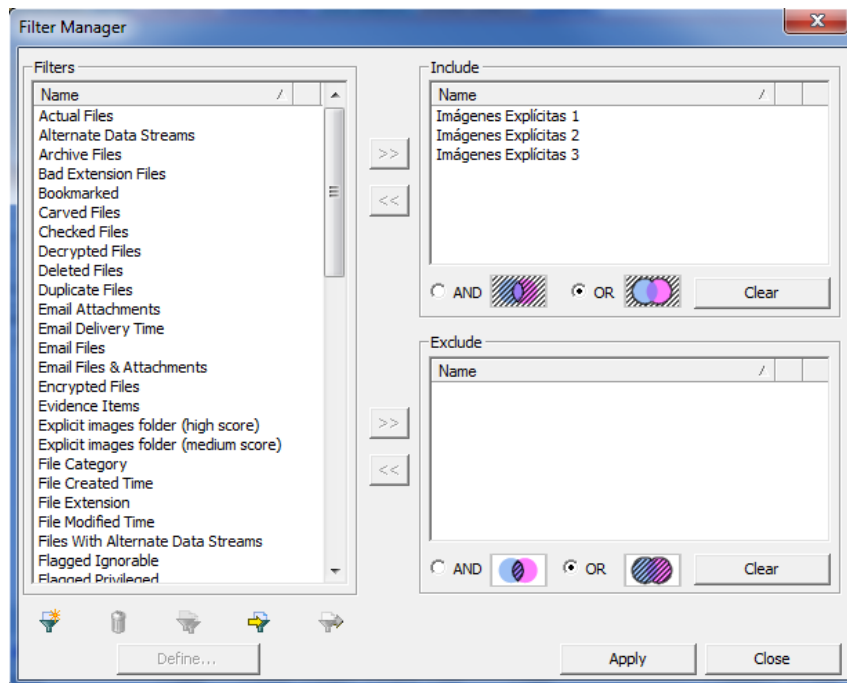


Fig. 4.16 Administración de los Filtros creados

Después de aplicar los filtros, en la ventana principal solamente se pueden observar las imágenes filtradas de un color amarillo, en la Figura 4.17 se puede ver un ejemplo al aplicar los filtros.

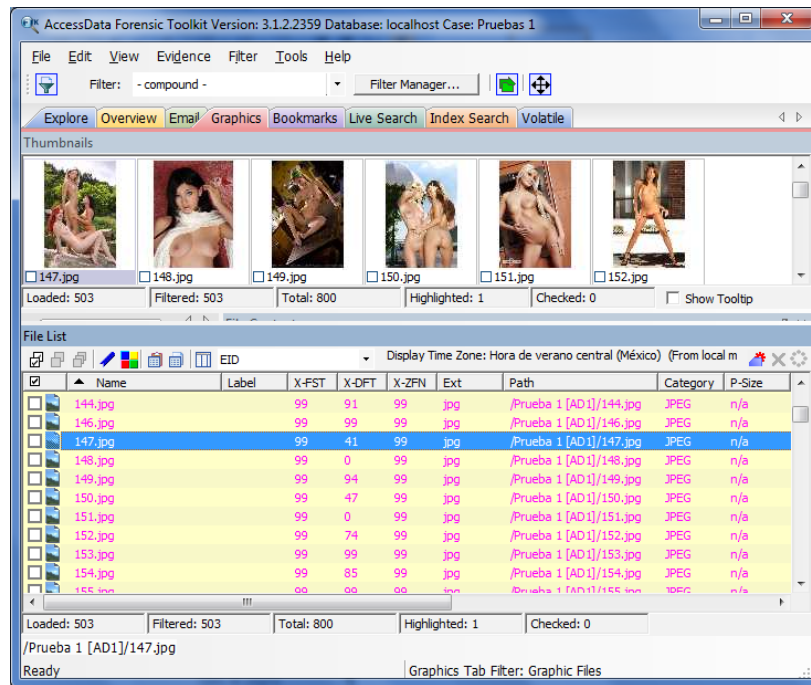


Fig. 4.17 Imágenes Filtradas

Después de filtrar las imágenes, para poder hacer una clasificación más sencilla se crea un marcador para poder separar las imágenes que realmente si son imágenes con contenido explícito y en otro marcador se colocan los falsos positivos. En la Figura 4.18 se muestra la pantalla donde se genera el marcador.

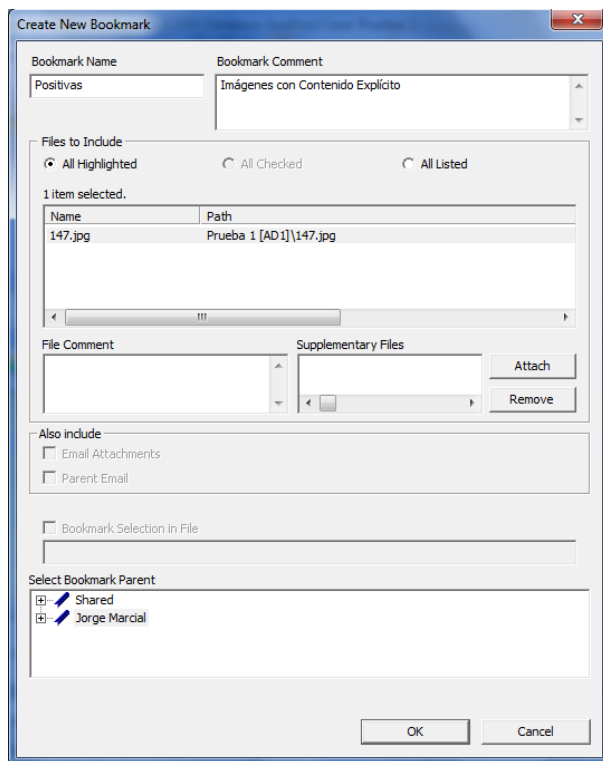


Fig. 4.18 Creación de un marcador

Este mismo procedimiento se hizo para cada uno de los tres conjuntos de prueba, para saber cuál es la tasa de reconocimiento y falsos positivos que puede alcanzar el sistema.

Al finalizar las pruebas con los tres conjuntos se obtuvo el siguiente resultado, el cual es mostrado en la Figura 4.19, que es una gráfica que muestra los porcentajes obtenidos tanto de reconocimiento como de falsos positivos y falsos negativos en cada uno de los conjuntos.

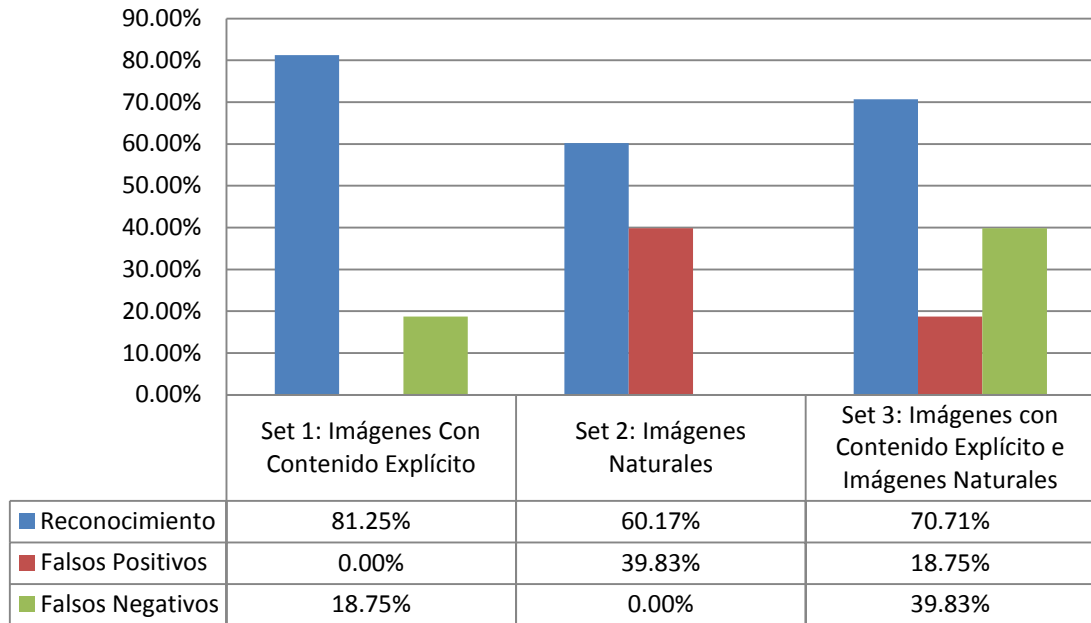


Fig. 4.19 Resultados obtenidos con FTK 3.1

Analizando los porcentajes se puede concluir que usando el modulo Explicit Image Detection del software Forensic Toolkit 3.1 se puede obtener un 81.25% de reconocimiento de las imágenes, con un 39.80% de falsos positivos en promedio y una tasa de falsos negativos del 18.75%.

4.4 Paraben's Porn Detection Stick

En el caso de este producto se hicieron diversas pruebas para el poder encontrar el mayor rango de reconocimiento, debido a que este software cuenta con una variable de sensibilidad la cual dependiendo su valor es la cantidad de imágenes encontradas positivamente, pero esto implica que existan también un mayor número de falsos positivos.

La Figura 4.20 muestra una gráfica de los porcentajes de reconocimiento y falsos positivos, usando el primer conjunto de imágenes y una serie de valores de sensibilidad para la detección.

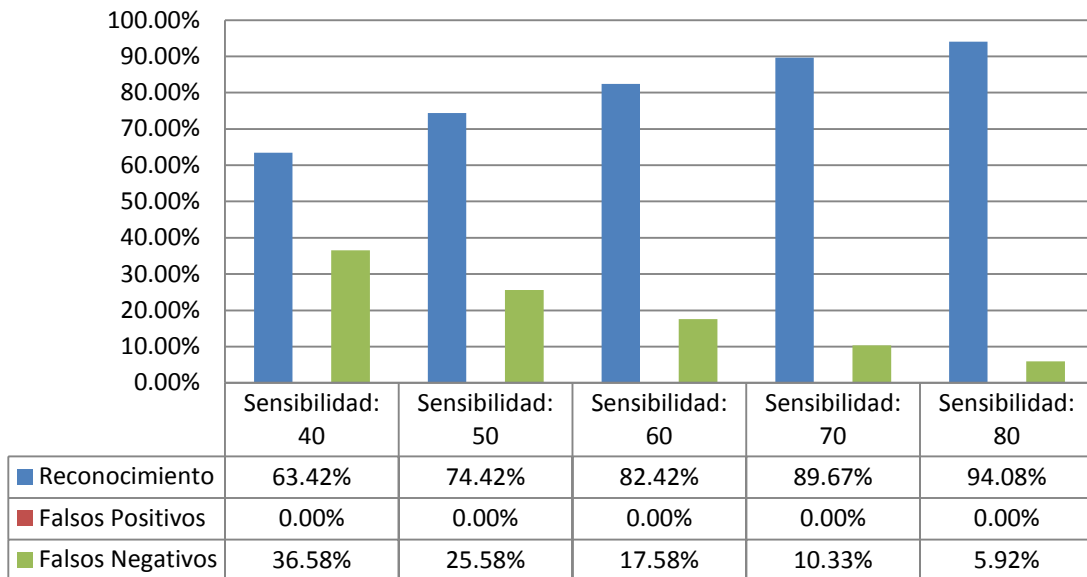


Fig. 4.20 Porcentajes de reconocimiento, falsos positivos y negativos del primer conjunto usando Paraben's Porn Detection Stick

En la figura anterior se puede observar que el mejor valor de sensibilidad se encuentra entre 60 y 70, que es donde se consigue un mayor porcentaje de reconocimiento sin tener un gran número de imágenes de falsos positivos.

En la Figura 4.21 se muestra el reporte generado usando una sensibilidad de valor 60.

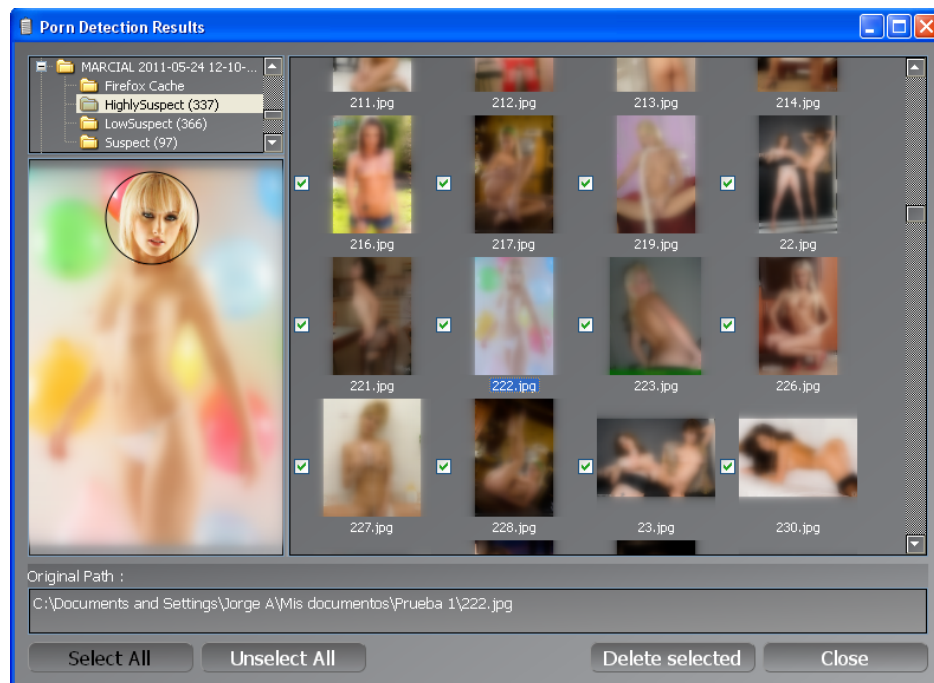


Fig. 4.21 Reporte del primer conjunto usando un valor de sensibilidad de 60

Para el segundo caso se hicieron las mismas pruebas usando las mismas sensibilidades que se utilizaron en la prueba anterior, el resultado de esta prueba se muestra en la Figura 4.22 que es una gráfica que muestra los porcentajes de dicha prueba.

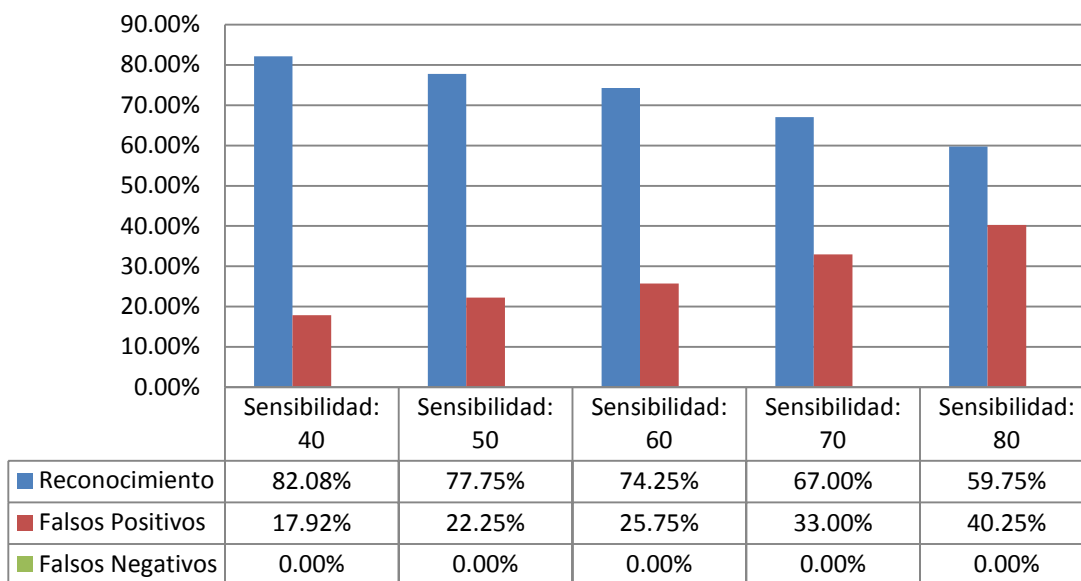


Fig. 4.22 Porcentajes de reconocimiento, falsos positivos y negativos del segundo conjunto de imágenes

En este caso se puede observar que igual que el experimento anterior el mayor porcentaje se logra usando un valor de sensibilidad de 60, en la Figura 4.23 se muestra el reporte que se generó cuando se hizo en la detección de imágenes.

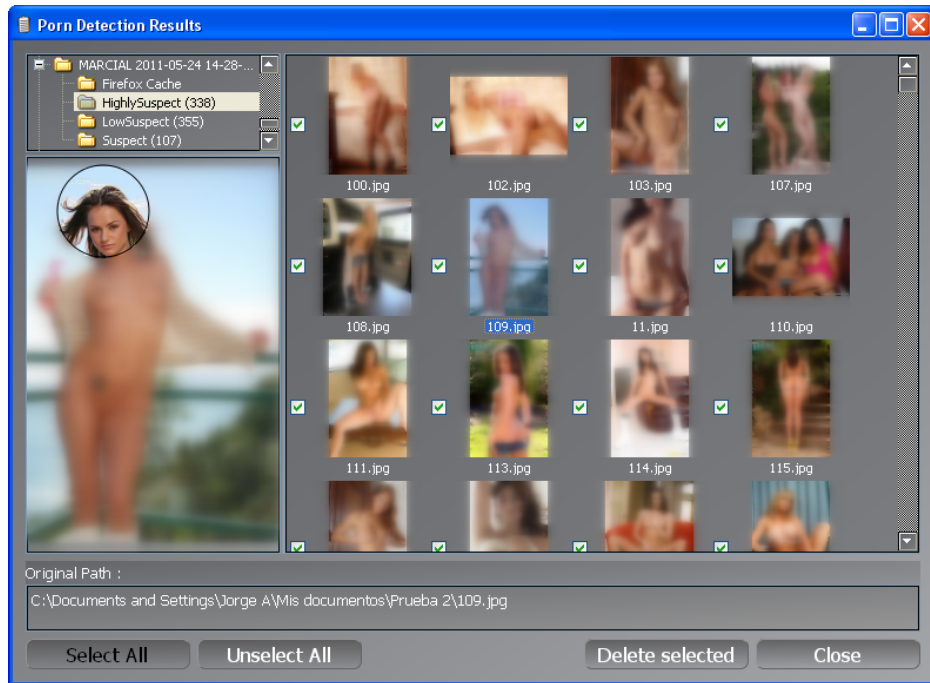


Fig. 4.23 Reporte del segundo conjunto de imágenes

En el último caso de estudio se hicieron las mismas pruebas que las de los dos conjuntos de imágenes anteriores, los resultados obtenidos de ésta prueba son mostrados en la Figura 4.24, la cual es una gráfica que muestra los porcentajes de reconocimiento y falsos positivos obtenidos usando distintos niveles de sensibilidad.

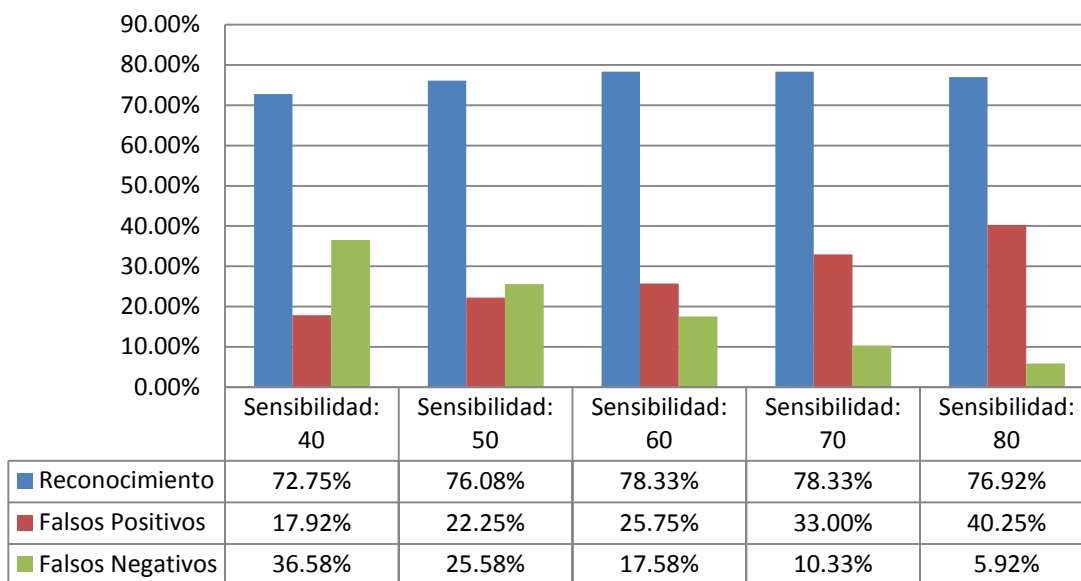


Fig. 4.24 Resultados del tercer conjunto de imágenes

En esta figura se puede observar que el mayor porcentaje obtenido se logró usando una sensibilidad igual a 70. El reporte generado de ésta prueba es mostrado en la Figura 4.25.

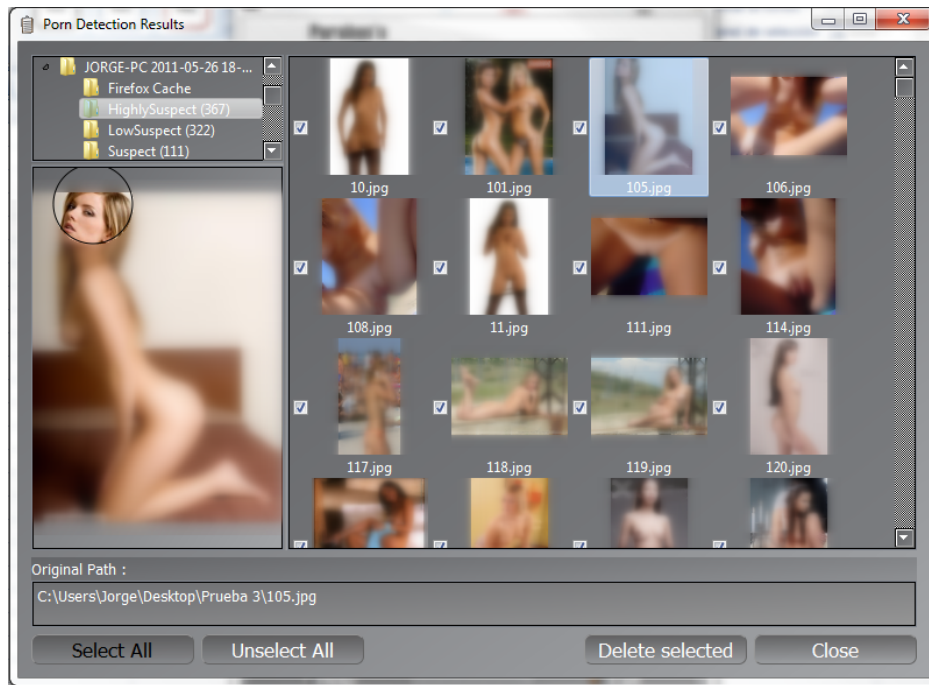


Fig. 4.25 Reporte generado del tercer conjunto de imágenes

Otro aspecto importante en la comparación de los resultados obtenidos es el porcentaje de Reconocimiento, en la Figura 4.26 se muestra una gráfica con este dato donde el porcentaje mostrado corresponde a los modelos de color utilizados para el sistema propuesto, en el caso de la herramienta FTK 3.1 es el que se obtuvo usando la combinación de los tres algoritmos y por último para la herramienta Paraben's Porn Detection Stick se tomó el dato correspondiente a una sensibilidad igual a 60.

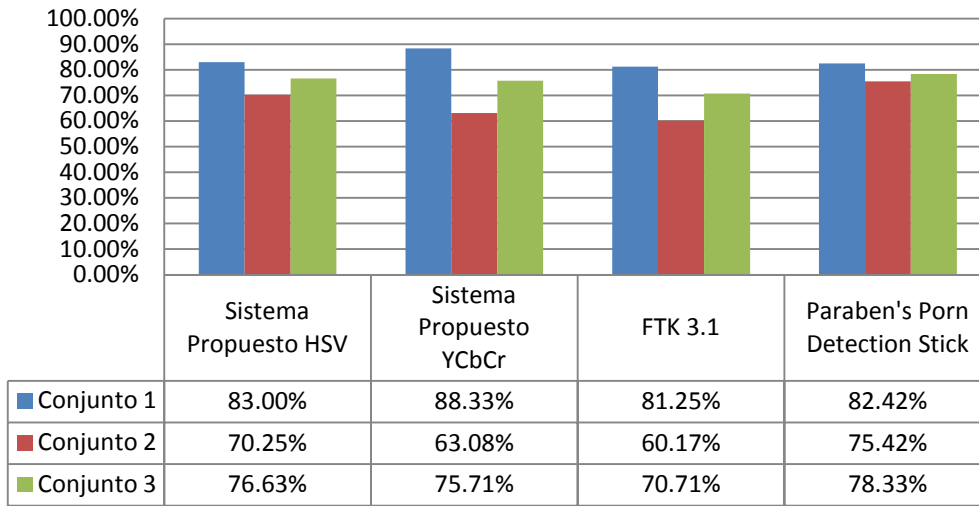


Fig. 4.26 Porcentaje de Reconocimiento

La Figura 4.27 y 4.28 muestran las gráficas con los porcentajes obtenidos de los Falsos Positivos y Falsos Negativos obtenidos por el sistema propuesto y los dos sistemas comerciales, esto con la finalidad hacer una comparativa entre dichos sistemas, y de esta forma saber cuál es el más eficaz.

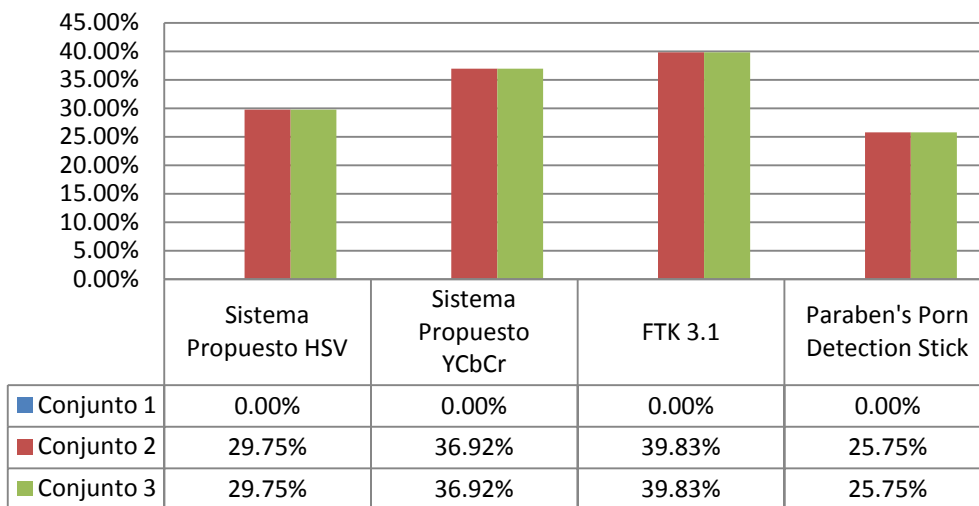


Fig. 4. 27 Porcentaje de Falsos Positivos

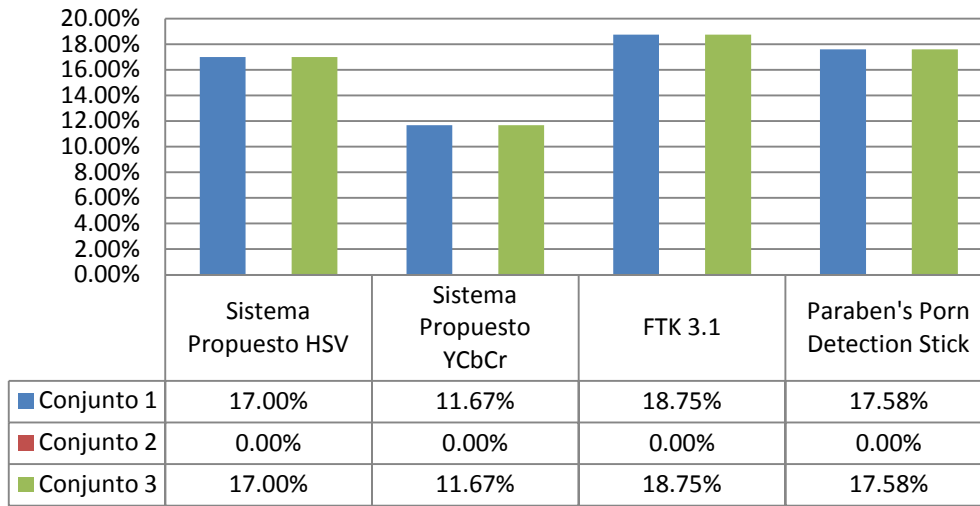


Fig. 4. 28 Porcentaje de Falsos Negativos

CONCLUSIONES

En este capítulo se presentan las conclusiones obtenidas durante el desarrollo de este trabajo.

5.1 Conclusiones Generales

La motivación para la realización de esta tesis han sido la elaboración de un algoritmo y una interfaz para el usuario final capaz de ayudar a la detección de imágenes pornográficas o con contenido explícito. Debido a la problemática actual que sufre nuestro país, ser uno de los mayores productores de pornografía infantil; en México no existe alguna alternativa creada por mexicanos para la investigación de este tipo de imágenes.

Los modelos de color HSV y YCbCr son importantes métodos para poder disminuir los problemas de iluminación que pueda tener la imagen analizada, esto se logra utilizando sólo las componentes de la crominancia Cb y Cr para el modelo YCbCr, en el otro caso se utilizan las tres componentes del modelo HSV.

Existen distintos factores que genera que algunos píxeles con tonos de piel sean detectados como no piel, entre estos factores se encuentran la calidad de la imagen; la poca iluminación con la que la imagen fue tomada; la sobre exposición de iluminación en algunas zonas de la piel, este problema es mayor en personas con un tono de piel claro; la modificación de las imágenes con software especializado, por ejemplo Adobe Photoshop.

Otro problema es la detección de píxeles que no contienen algún tono de piel, estos problemas se pueden encontrar principalmente en objetos de la naturaleza que tienen tonos muy parecidos a los de la piel humana, como por ejemplo: arboles, madera, arena, rocas, etc. Existen algunas imágenes las cuales fueron tomadas usando la iluminación generada en la puesta de sol, que hacen que cualquier objeto que aparezca en dichas imágenes sea detectado como piel debido a la iluminación antes mencionada.

El software FTK 3.1 es una herramienta que permite al usuario realizar el análisis de imágenes con contenido explícito usando tres algoritmos para poder determinar si la imagen analizada es o no una imagen clasificada como para adultos.

La herramienta Paraben's Porn Detection Stick permite al usuario buscar todas las imágenes localizadas en una computadora, las cuales son clasificadas en dos grupos, el primero llamado imágenes altamente sospechosas y el segundo llamado imágenes sospechosas, dependiendo de la clasificación que da el algoritmo utilizado por esta herramienta será donde se coloque la imagen analizada.

5.2 Logros del proyecto

Durante la elaboración de este trabajo se hicieron pruebas usando dos de los modelos de color presentados en los capítulos anteriores, en el primero de los casos se utilizó el modelo HSV como modelo para crear un clasificador de piel para poder discriminar todos aquellos píxeles que no contengan algún tono de piel, usando este modelo se pudo obtener un porcentaje de reconocimiento de 72.87% en uno de los conjuntos de imágenes con los que fue probado.

En el segundo de los casos se utilizó el modelo YCbCr como detector de piel para obtener todos los píxeles detectados como piel dentro de una imagen, usando este modelo se obtuvo un reconocimiento de 72.37% en el mejor de los casos.

La comparación entre los dos sistemas de sistemas propietarios y el sistema propuesto en este trabajo es con la finalidad de saber si éste puede trabajar de forma eficiente en el análisis de imágenes con contenido explícito,

y de esta manera poder utilizarlo como una herramienta para informática forense.

5.3 Trabajo a Futuro

Existen algunas consideraciones para poder hacer que el sistema pueda obtener un mayor porcentaje de reconocimiento, como crear un filtro de textura para poder discriminar los pixeles que correspondan a algún objeto de la naturaleza, crear un detector para la silueta del cuerpo para determinar la cantidad real de piel mostrada dentro de la imagen, la combinación de ambos sistemas podrían generar un mejor rendimiento del sistema.

Las imágenes en escala de grises no están tomadas en cuenta por ningún sistema tratado en esta tesis por lo que sería una buena investigación.

REFERENCIAS

- [1] Estadísticas sobre Disponibilidad y Uso de Tecnología de Información y Comunicaciones en los Hogares, 2009, INEGI.
- [2] (2011) Periódico El Universal [Online] Available
<http://www.eluniversal.com.mx/notas/743212.html>
- [3] (2011) Cyber Patrol Website. [Online]. Available
<http://www.cyberpatrol.com/>
- [4] (2011) Content Protect Website. [Online]. Available
<http://www.contentwatch.com/>
- [5] (2011) Netnanny Website. [Online]. Available
<http://www.netnanny.com/?pid=>
- [6] (2011) Family.Net Website. [Online]. Available <http://www.family.net>
- [7] (2011) K9 Web Protection Website. [Online]. Available
<http://www1.k9webprotection.com/>
- [8] (2011) Saint Lite 2.0 Website. [Online]. Available
<http://www.saintapp.com/es/>
- [9] (2011) Periódico El Economista. [Online]. Available
<http://eleconomista.com.mx/tecnociencia/2011/03/16/saint-software-mexicano-anti-porno>
- [10] M.M. Fleck, D.A. Forsyth, C. Bregler, “Finding naked people”, Proceedings of European Conference on Computer Vision, vol. 2, pp. 592–602, 1996.

- [11] Y. Dai, Y. Nakano, "Face-texture model based on SGLD and its application in face detection in a color scene", *Pattern Recognition* 29 (6), pp. 1007–1017, 1996.
- [12] K. Sobottka, I. Pitas, "A novel method for automatic face segmentation, facial feature extraction and tracking", *Signal Process Image Commun* 12, pp. 263–281, 1998.
- [13] D. Chai, K.N. Ngan, "Face segmentation using skin-color map in videophone applications", *IEEE Trans. on Circuits Syst. Video Technol.*, Vol.9, pp. 551 – 564, 1999.
- [14] C. Garcia, G. Tziritas, "Face detection using quantized skin color regions merging and wavelet packet analysis", *IEEE Trans. on Multimedia*, Vol. 1 (3), pp. 264–277, 1999.
- [15] Y. Wang, B. Yuan, "A novel approach for human face detection from color images under complex background", *Pattern Recognition* 34 (10), pp. 1983–1992, 2001.
- [16] H. Yao, W. Gao, "Face detection and location based on skin chrominance and lip chrominance transformation from color images", *Pattern Recognition* 34 (8), pp. 1555–1564, 2001.
- [17] K.W.Wong, K.M. Lam, W.C. Siu, "A robust scheme for live detection of human faces in color images", *Signal Process. Image Commun*, Vol. 18 (2), pp. 103–114, 2003.
- [18] F. Tomaz, T. Candeias, H. Shahbazkia, *Fast and accurate skin segmentation in color images*, CRV04, 2004.
- [19] C. Cusano, C. Brambilla, R. Schettini, G. Ciocca, "On the Detection of pornographic digital images", *VCIP*, pp. 2105-2113, 2003.

- [20] Y. Chan, R. Harvey, D. Smith, "Building systems to block pornography", In Challenge of Image Retrieval BCS Electronic Workshops in Computing series, 1999, pp 34-40.
- [21] Bosson, G.C. Cawley, Y. Chan, R. Harvey, "Non-Retrieval: blocking pornographic images", ACM CIVR Lecture Notes in Computer Science, Vol.2383, 2002, pages 60-69.
- [22] W. Kim, S.J. Yoo, J-s. Kim, T.Y. Nam, K. Yoon, "Detecting Adult Images Using Seven MPEG-7 Visual Descriptors", Web and Communication Technologies and Internet-Related Social Issues - HSI 2005, Vol. 3597, 2005, pages 336-339.
- [23] Q. Zhu, C-T. Wu, K-T. Cheng, Y-L. Wu, "An adaptive skin model and its application to objectionable image filtering", ACM Multimedia, 2004, pages 56-63.
- [24] W.Arentz, B.Olstad, "Classifying offensive sites based on image content", In Computer Vision and Image Understanding 2004, 2004, pp. 295-310.
- [25] QF.Zheng, MJ.Zhang, WQ.Wang, "A Hybrid Approach to Detect Adult Web Images", Advances in Multimedia Information Processing - PCM 2004, 2004, pp. 609-616.
- [26] J.-S. Lee, Y.-M. Kuo, P.-C. Chung, E.-L. Chen, "Naked image detection based on adaptive and extensible skin colour model", Journal Pattern Recognition, Vol. 40 Issue 8, 2007, pp. 2261 – 2270.
- [27] W. Kelly, A. Donnellan and D. Molloy, "Screening for Objectionable Images: A Review of Skin Detection Techniques", In Machine Vision and Image Processing Conference (IMVIP '08), pp. 151 – 158, 2008.

- [28] P. Kakumanu, S. Makrogiannis, N. Bourbakis, "A survey of skin-color modeling and detection methods", In *Pattern Recognition*, Vol. 40, pp. 1106-1122, 2007.
- [29] Rafael C. Gonzalez y Richard E. Woods, *Digital Image Processing*, 2nd ed., Ed. Prentice Hall, 2002.
- [30] V. Vezhnevets, V. Sazonov, A. Andreeva, "A Survey on Pixel-based Skin Color Detection Techniques", In Proceedings of the GrapiCon, pp. 85-92, 2003.
- [31] Nick Efford, *Digital Image Processing a practical introduction using JAVA*, 1st ed., Ed. Addison-Wesley, 2000.
- [32] Keith Jack, *Video Demystified: A Handbook for the Digital Engineer*, 4th Edition, Ed. Elseiver, 2005.
- [33] Peer P., Kovac J., AND Solina F., "Human skin colour clustering for face detection", In *submitted to EUROCON 2003 – International Conference on Computer as a Tool*, vol. 2, pp. 144-148, 2003.
- [34] Jones M. J., y Rehg J. M., "Statistical color models with application to skin detection", In Proc. of the CVPR '99, vol. 1, 274–280, 1999.
- [35] Zarit B. D., Super B. J., y Quek F. K. H., "Comparison of five color models in skin pixel classification", In ICCV'99 Int'l Workshop on recognition, analysis and tracking of faces and gestures in Real-Time systems, pp. 58–63, 1999.
- [36] B. Martín del Brío y Alfredo Sanz Molina, *Redes Neuronales y Sistemas Difusos*, 2^a. Edición, Ed. Alfaomega, 2002.
- [37] Forensic Toolkit 3.1 User Guide.

GLOSARIO

- **Crominancia**

Es la componente que contiene la información de color de una imagen.

- **Falso Negativo**

Para este caso en particular un falso negativo es cuando el sistema detecta una imagen con contenido explícito como una imagen natural.

- **Falso Positivo**

En este caso un falso positivo es cuando el sistema reconoce una imagen natural como una imagen con contenido explícito.

- **Histograma**

Es una representación gráfica de una variable en forma de barras. En el eje vertical se representa la frecuencia con la que aparece un valor de esa variable; en el eje horizontal los valores de la variable son representados.

- **Imagen Digital**

Es una representación bidimensional de una imagen, la cual puede considerarse una matriz donde cada elemento identifica a un punto en particular de la imagen.

- **Imagen Forense**

Es un archivo o un dispositivo que contiene la estructura y contenido completo de un dispositivo o medio de almacenamiento como: disco duro, unidades flash USB, DVD, etc. Una imagen forense se crea haciendo una copia sector por sector o bit a bit, del dispositivo de almacenamiento.

- **Imagen Natural**

Es aquella que por su naturaleza no contiene contenido explícito o pornográfico, por ejemplo: Personas vestidas, animales, plantas, automóviles, dibujos animados, paisajes, etc.

- **Informática Forense**

Es la ciencia o arte de preservar, capturar, procesar y analizar datos de una computadora usando una metodología, en donde toda evidencia encontrada puede ser presentada, tanto para prosecución judicial como para una sanción interna de una organización.

- **Luminancia**

Es la componente que contiene la información de luz o brillo de una imagen.

- **NTSC**

NTSC (National Television System Comitee) o Comisión Nacional de Sistema de Televisión, es un sistema de codificación y transmisión para la televisión analógica en color, el cual es utilizado en la mayor parte de América y Japón.

- **PAL**

PAL (Phase Alternating Line) o Línea de Fase Alternada, es el nombre con el que se designa al sistema de codificación utilizado en la transmisión de señales de televisión analógica en color, utilizada en la mayor parte del mundo.

- **Píxel**

Es la abreviatura de “picture element”, el cual es el elemento de una imagen digital.

- **Procesamiento Digital de Imágenes**

Es la aplicación de una serie de algoritmos especializados para procesar imágenes digitales. Los resultados de este proceso pueden ser imágenes o un conjunto de características representativas o propiedades de las imágenes originales. Las aplicaciones del procesamiento digital de imágenes han sido comunes en robótica, sistemas inteligentes, imágenes médicas, detección a distancia, y forense.

- **Sistema de coordenadas cartesiano**

El sistema está constituido por un plano, el cual consta de dos rectas perpendiculares que se intersectan en un punto al que se llama **origen**. La recta horizontal se le da el nombre de eje X o eje de las abscisas; la otra recta, en posición vertical se le denomina eje Y o eje de las ordenadas.

Es la medida de la cantidad de energía que un observador percibe procedente de una fuente luminosa, ésta se mide en lúmenes.

- **Umbral**

Es el valor mínimo de una variable a partir del cual tiene lugar un determinado efecto.

ANEXO A: PROGRAMAS

Interfaz Gráfica Principal

```

function varargout = EID_final(varargin)
% EID_FINAL M-file for EID_final.fig
%     EID_FINAL, by itself, creates a new EID_FINAL or raises the
existing
%     singleton*.
%
%     H = EID_FINAL returns the handle to a new EID_FINAL or the
handle to
%     the existing singleton*.
%
%     EID_FINAL('CALLBACK', hObject,eventData,handles,...) calls the
local
%     function named CALLBACK in EID_FINAL.M with the given input
arguments.
%
%     EID_FINAL('Property','Value',...) creates a new EID_FINAL or
raises the
%     existing singleton*. Starting from the left, property value
pairs are
%     applied to the GUI before EID_final_OpeningFunction gets
called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to EID_final_OpeningFcn via
varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help EID_final

% Last Modified by GUIDE v2.5 05-May-2011 16:52:21

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @EID_final_OpeningFcn, ...
                  'gui_OutputFcn',  @EID_final_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

```

```

if narginout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before EID_final is made visible.
function EID_final_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to EID_final (see VARARGIN)

% Choose default command line output for EID_final
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes EID_final wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% En esta sección se cargan los escudos del IPN y de ESIME
clc
axes(handles.axes1)
axis off
ipn=imread('logo_ipn.jpg');
imshow(ipn)
axes(handles.axes2)
axis off
esime=imread('logo_esime.jpg');
imshow(esime)

% --- Outputs from this function are returned to the command line.
function varargout = EID_final_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)

```

```
% hObject    handle to pushbutton1 (see GCBO) % eventdata  reserved
- to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Este botón abre la interfaz para el análisis de una sola imagen
delete(handles.figure1)
Analisis_imagen_final

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Este botón abre la interfaz para el análisis de un directorio
delete(handles.figure1)
Analisis_directorio_final

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Este botón genera la pregunta cuando se desea salir de la interfaz
resp=questdlg('¿Desea salir del programa?', 'Detección de
Imágenes', 'Si', 'No', 'Cancelar');
if strcmp(resp, 'No')
    return;
else
    clear,clc,close all
end
```


Interfaz para el análisis de una imagen

```

function varargout = Analisis_imagen_final(varargin)
% ANALISIS_IMAGEN_FINAL M-file for Analisis_imagen_final.fig
%     ANALISIS_IMAGEN_FINAL, by itself, creates a new
ANALISIS_IMAGEN_FINAL or raises the existing
%     singleton*.
%
%     H = ANALISIS_IMAGEN_FINAL returns the handle to a new
ANALISIS_IMAGEN_FINAL or the handle to
%     the existing singleton*.
%
%
ANALISIS_IMAGEN_FINAL('CALLBACK',hObject,eventData,handles,...)
calls the local
%     function named CALLBACK in ANALISIS_IMAGEN_FINAL.M with the
given input arguments.
%
%     ANALISIS_IMAGEN_FINAL('Property','Value',...) creates a new
ANALISIS_IMAGEN_FINAL or raises the
%     existing singleton*. Starting from the left, property value
pairs are
%     applied to the GUI before Analisis_imagen_final_OpeningFcn
gets called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to
Analisis_imagen_final_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help
Analisis_imagen_final

% Last Modified by GUIDE v2.5 16-Nov-2011 12:11:52

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn',
@Analisis_imagen_final_OpeningFcn, ...
                  'gui_OutputFcn',
@Analisis_imagen_final_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

```

```
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Analisis_imagen_final is made visible.
function Analisis_imagen_final_OpeningFcn(hObject, eventdata,
handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Analisis_imagen_final (see
VARARGIN)

% Choose default command line output for Analisis_imagen_final
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes Analisis_imagen_final wait for user response (see
UIRESUME)
% uiwait(handles.figure1);
clc
clear global img
clear global dir_save
axes(handles.axes1)
axis off
ipn=imread('logo_ipn.jpg');
imshow(ipn)
axes(handles.axes2)
axis off
esime=imread('logo_esime.jpg');
imshow(esime)
axes(handles.axes3)
axis off
axes(handles.axes4)
axis off

% --- Outputs from this function are returned to the command line.
function varargout = Analisis_imagen_final_OutputFcn(hObject,
eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global img
[FileName,PathName,FilterIndex]=uigetfile({'*.jpg'; '*.bmp'; '*.png'},
'Select an Image');
img=strcat(PathName,FileName);
axes(handles.axes3)
imagen=imread(img);
imshow(imagen)

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global img
% global dir_save
if length(img)==0
    errordlg('Debe de seleccionar una imagen','Detección de
Imágenes');
else
    if(get(handles.radiobutton1,'value')==0 &&
get(handles.radiobutton2,'value')==0)
        errordlg('Debe de seleccionar una opción','Detección de
Imágenes');
    else
        %Selección del Modo de operación
        if (get(handles.radiobutton1,'value')==1)
%
            op=BusquedaHSV2(img);
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

            clear imp
            im=imread(img);
            x=im(:,:,1);
            y=im(:,:,2);
            z=im(:,:,3);
            %Bloque de Transformación
            imagen=rgb2hsv(im);
            H=imagen(:,:,1);%H
            S=imagen(:,:,2);%S
            V=imagen(:,:,3);%V
            [m n]=size(x);
            %Matrices nuevas
            x1=zeros(m,n);
            r=zeros(m,n);

```

```

g=zeros(m,n);
b=zeros(m,n);
%Bloque de detección de piel
for i=1:m
    for j=1:n
        if H(i,j)<=0.12 && S(i,j)>0.15 && S(i,j)<0.9 &&
V(i,j)>0.1 && V(i,j)<0.95
            x1(i,j)=1;
        end
    end
end
se = strel('disk',5);
I_opened = imopen(x1,se);
P = bwlabeln(I_opened);
imagen1 = label2rgb(P);
%Contar el número de pixeles de conectados por zonas
for t=1:max(max(P))
    cont=0;
    for i=1:m
        for j=1:n
            if P(i,j)==t
                cont=cont+1;
            end
            contador(t)=cont;
        end
    end
end
for t=1:max(max(P))
    aux=contador(t)/max(contador);
    if aux<0.03 && contador(t)~=0
        I_opened(P == t)=0;
        contador(t)=0;
    end
end

%
end

for i=1:m
    for j=1:n
        if I_opened(i,j)==1
            r(i,j)=x(i,j);
            g(i,j)=y(i,j);
            b(i,j)=z(i,j);
        end
    end
end
imp(:,:,1)=r;
imp(:,:,2)=g;
imp(:,:,3)=b;
imp=uint8(imp);
final{1,1}=imp;
if isempty(contador)
    pielfinal=0;
else
    pielfinal=max(contador)/sum(contador)*100;
end

```

```

end
    cont2=0;
for i=1:m
    for j=1:n
        if I_opened(i,j)~=0;
            cont2=cont2+1;
        end
    end
end
porcen_piel=(cont2/(m*n))*100;

final{2,1}=pielfinal;
resultado=final;
axes(handles.axes4)
imshow(uint8(resultado{1,1}))
aux=num2str(resultado{2,1});
porcentaje=strcat(aux,'%');
if resultado{2,1}>58 && porcen_piel>18
    set(handles.text8,'String',porcentaje);
    set(handles.text9,'String','Imagen con Contenido
Explícito');
else
    set(handles.text8,'String',porcentaje);
    set(handles.text9,'String','Imagen sin Contenido
Explícito');
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

elseif (get(handles.radiobutton2,'value')==1)
%
    op=BusquedaYCbCr2(img);
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    clear imp
    im=imread(img);
    x=im(:,:,1);
    y=im(:,:,2);
    z=im(:,:,3);
    %Bloque de Transformación
    imagen=rgb2ycbcr(im);
    Y=imagen(:,:,1);%Y
    Cb=imagen(:,:,2);%Cb
    Cr=imagen(:,:,3);%Cr
    [m n]=size(x);
    %Matrices nuevas
    x1=zeros(m,n);
    r=zeros(m,n);
    g=zeros(m,n);
    b=zeros(m,n);
    %Bloque de detección de piel
    for i=1:m
        for j=1:n
            if Cb(i,j)>=80 && Cb(i,j)<=120 && Cr(i,j)>=133
&& Cr(i,j)<=173
                x1(i,j)=1;
            end
        end
    end
end

```

```

        end
    end
    se = strel('disk',5);
    I_opened = imopen(x1,se);
    P = bwlabeln(I_opened);

    for t=1:max(max(P))
        cont=0;
        for i=1:m
            for j=1:n
                if P(i,j)==t
                    cont=cont+1;
                end
                contador(t)=cont;
            end
        end
    end

    %Eliminar las zonas con menos de 5000 pixeles conectados

    for t=1:max(max(P))
        aux=contador(t)/max(contador);
        if aux<0.1 && contador(t)~=0
            I_opened(P == t)=0;
            contador(t)=0;
        end
    end

    for i=1:m
        for j=1:n
            if I_opened(i,j)==1
                r(i,j)=x(i,j);
                g(i,j)=y(i,j);
                b(i,j)=z(i,j);
            end
        end
    end

    imp(:,:,1)=r;
    imp(:,:,2)=g;
    imp(:,:,3)=b;
    imp=uint8(imp);
    final{1,1}=imp;

    if isempty(contador)
        pielfinal=0;
    else
        pielfinal=max(contador)/sum(contador)*100;
    end
    %Bloque de Cuantificación
    cont2=0;
    for i=1:m
        for j=1:n
            if I_opened(i,j)~=0;

```

```

        cont2=cont2+1;
    end
end
end
porcen_piel=(cont2/(m*n))*100;
final{2,1}=pielfinal;
resultado=final;
axes(handles.axes4)
imshow(uint8(resultado{1,1}))
aux=num2str(resultado{2,1});
porcentaje=strcat(aux, '%');
if resultado{2,1}>58 && porcen_piel>18
    set(handles.text8, 'String', porcentaje);
    set(handles.text9, 'String', 'Imagen con Contenido
Explícito');
else
    set(handles.text8, 'String', porcentaje);
    set(handles.text9, 'String', 'Imagen sin Contenido
Explícito');
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
end
end
end

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
clear global img
set(handles.radiobutton1, 'value', 0);
set(handles.radiobutton2, 'value', 0);
set(handles.text8, 'String', '');
set(handles.text9, 'string', '');
axes(handles.axes3)
imshow(255)
axes(handles.axes4)
imshow(255)
msgbox('Seleccione de nuevo todos los parámetros', 'Detección de
Imágenes ');

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
delete(handles.figure1)
EID_final

```

```
% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
resp=questdlg('¿Desea salir del programa?', 'Detección de
Imágenes', 'Si', 'No', 'Cancelar');
if strcmp(resp, 'No')
    return;
else
    clear,clc,close all
end

% --- Executes on button press in radiobutton1.
function radiobutton1_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of radiobutton1
set(handles.radiobutton1, 'value',1);
set(handles.radiobutton2, 'value',0);

% --- Executes on button press in radiobutton2.
function radiobutton2_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of radiobutton2
set(handles.radiobutton1, 'value',0);
set(handles.radiobutton2, 'value',1);
```


Interfaz para el análisis de un directorio

```

function varargout = Analisis_directorio_final(varargin)
% ANALISIS_DIRECTORIO_FINAL M-file for Analisis_directorio_final.fig
%     ANALISIS_DIRECTORIO_FINAL, by itself, creates a new
ANALISIS_DIRECTORIO_FINAL or raises the existing
%     singleton*.
%
%     H = ANALISIS_DIRECTORIO_FINAL returns the handle to a new
ANALISIS_DIRECTORIO_FINAL or the handle to
%     the existing singleton*.
%
%
ANALISIS_DIRECTORIO_FINAL('CALLBACK',hObject,eventData,handles,...)
calls the local
%     function named CALLBACK in ANALISIS_DIRECTORIO_FINAL.M with
the given input arguments.
%
%     ANALISIS_DIRECTORIO_FINAL('Property','Value',...) creates a
new ANALISIS_DIRECTORIO_FINAL or raises the
%     existing singleton*. Starting from the left, property value
pairs are
%     applied to the GUI before
Analisis_directorio_final_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to
Analisis_directorio_final_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help
Analisis_directorio_final

% Last Modified by GUIDE v2.5 16-Nov-2011 13:10:07

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn',  'gui_OpeningFcn',
@Analisis_directorio_final_OpeningFcn, ...
                  'gui_OutputFcn',  'gui_OutputFcn',
@Analisis_directorio_final_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

```

```
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Analisis_directorio_final is made
visible.
function Analisis_directorio_final_OpeningFcn(hObject, eventdata,
handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Analisis_directorio_final
(see VARARGIN)

% Choose default command line output for Analisis_directorio_final
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes Analisis_directorio_final wait for user response (see
UIRESUME)
% uiwait(handles.figure1);
clc
clear global ruta_dir
clear global dir_save
axes(handles.axes1)
axis off
ipn=imread('logo_ipn.jpg');
imshow(ipn)
axes(handles.axes2)
axis off
esime=imread('logo_esime.jpg');
imshow(esime)
set(handles.radiobutton1,'value',0);
set(handles.radiobutton2,'value',0);
set(handles.text5,'String','');
set(handles.text8,'string','');
set(handles.text9,'string','');
set(handles.checkbox1,'value',0);
set(handles.checkbox2,'value',0);
set(handles.checkbox3,'value',0);
set(handles.checkbox3,'value',0);

% --- Outputs from this function are returned to the command line.
```

```

function varargout = Analisis_directorio_final_OutputFcn(hObject,
eventdata, handles)
% varargout    cell array for returning output args (see VARARGOUT);
% hObject     handle to figure
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject     handle to pushbutton1 (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)
global ruta_dir
ruta_dir = uigetdir();

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject     handle to pushbutton2 (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

global ruta_dir
global dir_save
if length(ruta_dir)==0
    errordlg('Debe de seleccionar una carpeta','Detección de
Imágenes');
else if length(dir_save)==0
    errordlg('Seleccione una carpeta para guardar','Detección de
Imágenes');
else
    if(get(handles.radiobutton1,'value')==0 &&
get(handles.radiobutton2,'value')==0)
        errordlg('Debe de seleccionar una opción','Detección de
Imágenes');
    else
        %Selección del Modo de operación
        if (get(handles.radiobutton1,'value')==1)
%
            op=BusquedaHSV(ruta_dir,dir_save);
            %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Seleccionamos sólo los archivos jpg
archivos=[];
p=1;
if(get(handles.checkbox1,'value')==1
    %Cargamos los archivos jpg
    directorio1=strcat(ruta_dir,'/*.jpg');
    files1=dir(directorio1);
    for i=1:length(files1)

```

```

        im_total=files1(i,1).name;
        archivos{p}=im_total;
        p=p+1;
    end
end
if(get(handles.checkbox2,'value')==1
    directorio2=strcat(ruta_dir,'/*.bmp');
    files2=dir(directorio2);
    for i=1:length(files2)
        im_total=files2(i,1).name;
        archivos{p}=im_total;
        p=p+1;
    end
end
if(get(handles.checkbox3,'value')==1
    directorio3=strcat(ruta_dir,'/*.png');
    files3=dir(directorio3);
    for i=1:length(files3)
        im_total=files3(i,1).name;
        archivos{p}=im_total;
        p=p+1;
    end
end
%Medimos la longitud de la variable para saber
cuantas imágenes son
tam=length(archivos);
if tam==0
    errordlg({'La Carpeta NO CONTIENE IMÁGENES',
'Seleccione otro Directorio'}, 'Detección de imágenes');
    resultado={'-1'};
else
    l=0;
    %Se Analizan todas las Imágenes
    for k=1:tam
%       set(handles.text9,'String',k);
        clear imp
        nombreimagen=archivos{k};
dir_imagen=strcat(ruta_dir,'\',nombreimagen);
        im=imread(dir_imagen);

        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        x=im(:,:,1);
        y=im(:,:,2);
        z=im(:,:,3);
        %Bloque de Transformación
        imagen=rgb2hsv(im);
        H=imagen(:,:,1);%H
        S=imagen(:,:,2);%S
        V=imagen(:,:,3);%V
        [m n]=size(x);
        %Matrices nuevas
        x1=zeros(m,n);

```

```

r=zeros(m,n);
g=zeros(m,n);
b=zeros(m,n);
%Bloque de detección de piel
for i=1:m
    for j=1:n
        if H(i,j)<=0.12 && S(i,j)>0.15 &&
S(i,j)<0.9 && V(i,j)>0.1 && V(i,j)<0.95
            x1(i,j)=1;
        end
    end
end
se = strel('disk',5);
I_opened = imopen(x1,se);
P = bwlabeln(I_opened);
%
imagen1 = label2rgb(P);
%Contar el número de pixeles de conectados
por zonas

for t=1:max(max(P))
    cont=0;
    for i=1:m
        for j=1:n
            if P(i,j)==t
                cont=cont+1;
            end
            contador(t)=cont;
        end
    end
end
for t=1:max(max(P))
    aux=contador(t)/max(contador);
    if aux<0.03 && contador(t)~=0
        I_opened(P == t)=0;
        contador(t)=0;
    end
end

for i=1:m
    for j=1:n
        if I_opened(i,j)==1
            r(i,j)=x(i,j);
            g(i,j)=y(i,j);
            b(i,j)=z(i,j);
        end
    end
end
imp(:,:,1)=r;
imp(:,:,2)=g;
imp(:,:,3)=b;
imp=uint8(imp);
final{1,1}=imp;
if isempty(contador)
    pielfinal=0;
end

```

```

else
pielfinal=max(contador)/sum(contador)*100;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Bloque de Cuantificación
cont2=0;
for i=1:m
    for j=1:n
        if I_opened(i,j)~=0;
            cont2=cont2+1;
        end
    end
end
porcen_piel=(cont2/(m*n))*100;

final{k,1}=nombreimagen;
final{k,2}=pielfinal;
% disp(k)
if pielfinal>=58
    if porcen_piel > 18
        l=l+1;
        if l==1
            %SE crea una carpeta
            fecha=date;
            time=clock;
            tiempo1=num2str(time(4));
            tiempo2=num2str(time(5));
            tiempofinal=strcat(tiempo1,'-
',tiempo2);
            nombredir=strcat(fecha,'-
',tiempo2);
nombrefinal=strcat(dir_save, '\\',nombredir);
            mkdir(nombrefinal);
            end
            res(1,1)=tam;
            res(1,2)=porcen_piel;

nombre=strcat(nombrefinal, '\\',nombreimagen);
            imwrite(im,nombre);
            end
            end
            set(handles.text9, 'String',k);
            pause(0.03125)
end
if l==0
    resultado={num2str(l);num2str(tam)};
else
resultado={num2str(l);num2str(tam);nombrefinal};
end

```

```

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

elseif (get(handles.radiobutton2,'value')==1)
%
op=BusquedaYCbCr(ruta_dir,dir_save);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %Seleccionamos sólo los archivos jpg
    archivos=[];
    p=1;
    if(get(handles.checkbox1,'value')==1
        %Cargamos los archivos jpg
        directorio1=strcat(ruta_dir,'/*.jpg');
        files1=dir(directorio1);
        for i=1:length(files1)
            im_total=files1(i,1).name;
            archivos{p}=im_total;
            p=p+1;
        end
    end
    if(get(handles.checkbox2,'value')==1
        directorio2=strcat(ruta_dir,'/*.bmp');
        files2=dir(directorio2);
        for i=1:length(files2)
            im_total=files2(i,1).name;
            archivos{p}=im_total;
            p=p+1;
        end
    end
    if(get(handles.checkbox3,'value')==1
        directorio3=strcat(ruta_dir,'/*.png');
        files3=dir(directorio3);
        for i=1:length(files3)
            im_total=files3(i,1).name;
            archivos{p}=im_total;
            p=p+1;
        end
    end
    %Medimos la longitud de la variable para saber
    cuantas imágenes son
    tam=length(archivos);
    if tam==0
        errordlg({'La Carpeta NO CONTIENE IMÁGENES',
'Seleccione otro Directorio'},'Detección de imágenes');
        resultado={'-1'};
    else
        %SI Existen imágenes se crea una carpeta
        l=0;
        %Se Analizan todas las Imágenes
        for k=1:tam
            clear imp
            nombreimagen=archivos{k};

```

```

dir_imagen=strcat(ruta_dir, '\', nombreimagen);
im=imread(dir_imagen);
x=im(:,:,1);
y=im(:,:,2);
z=im(:,:,3);
%Bloque de Transformación
imagen=rgb2ycbcr(im);
Y=imagen(:,:,1);%Y
Cb=imagen(:,:,2);%Cb
Cr=imagen(:,:,3);%Cr
[m n]=size(x);
%Matrices nuevas
x1=zeros(m,n);
r=zeros(m,n);
g=zeros(m,n);
b=zeros(m,n);
%Bloque de detección de piel
for i=1:m
    for j=1:n
        if Cb(i,j)>=80 && Cb(i,j)<=120
&& Cr(i,j)>=133 && Cr(i,j)<=173
            x1(i,j)=1;
        end
    end
end
se = strel('disk',5);
I_opened = imopen(x1,se);
P = bwlabeln(I_opened);

for t=1:max(max(P))
    cont=0;
    for i=1:m
        for j=1:n
            if P(i,j)==t
                cont=cont+1;
            end
            contador(t)=cont;
        end
    end
end

%Eliminar las zonas con menos de 5000
pixeles conectados

for t=1:max(max(P))
    aux=contador(t)/max(contador);
    if aux<0.1 && contador(t)~=0
        I_opened(P == t)=0;
        contador(t)=0;
    end
end

for i=1:m

```



```

        for j=1:n
            if I_opened(i,j)==1
                r(i,j)=x(i,j);
                g(i,j)=y(i,j);
                b(i,j)=z(i,j);
            end
        end
    end
    imp(:,:,1)=r;
    imp(:,:,2)=g;
    imp(:,:,3)=b;
    imp=uint8(imp);
    if isempty(contador)
        pielfinal=0;
    else
        pielfinal=max(contador)/sum(contador)*100;
    end

    %Bloque de Cuantificación
    cont2=0;
    for i=1:m
        for j=1:n
            if I_opened(i,j)~=0;
                cont2=cont2+1;
            end
        end
    end
    porcen_piel=(cont2/(m*n))*100;

    final{k,1}=nombreimagen;
    final{k,2}=pielfinal;
    disp(k)
    if porcen_piel>=58
        if porcen_piel > 18
            l=l+1;
            if l==1
                %Se crea una carpeta
                fecha=date;
                time=clock;
                tiempo1=num2str(time(4));
                tiempo2=num2str(time(5));

                tiempofinal=strcat(tiempo1,'-',tiempo2);
                nombredir=strcat(fecha,'-',
                ',tiempofinal);

                nombrefinal=strcat(dir_save,'\ ',nombredir);
                mkdir(nombrefinal);
            end
            res(1,1)=tam;
            res(1,2)=porcen_piel;

            nombre=strcat(nombrefinal,'\ ',nombreimagen);

```

```

        imwrite(im,nombre);
    end
    end
    set(handles.text9,'String',k);
    pause(0.03125)
end
if l==0
    resultado={num2str(l);num2str(tam)};
else
resultado={num2str(l);num2str(tam);nombrefinal};
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

end
if strcmp(resultado{1,1},'-1')==1
    set(handles.radiobutton1,'value',0);
    set(handles.radiobutton2,'value',0);
else
    if resultado{1,1}=='0'
        detectadas=strcat(resultado{1,1},' ','
imágenes');
        set(handles.text5,'String',detectadas);
        set(handles.text8,'string','La carpeta no fue
creada');
    else
        if strcmp(resultado{1,1},'1')==1
            detectadas=strcat(resultado{1,1},' ','
imagen');
            set(handles.text5,'String',detectadas);
            dirnuevo=resultado{3,1};
            set(handles.text8,'string',dirnuevo);
        else
            detectadas=strcat(resultado{1,1},' ','
imágenes');
            set(handles.text5,'String',detectadas);
            dirnuevo=resultado{3,1};
            set(handles.text8,'string',dirnuevo);
        end
    end
end
end
end
end

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
resp=questdlg('¿Desea salir del programa?','Detección de
Imágenes','Si','No','Cancelar');

```

```

if strcmp(resp, 'No')
    return;
else
    clear,clc,close all
end

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global dir_save
dir_save = uigetdir();

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
clear global ruta_dir
clear global dir_save
set(handles.radiobutton1, 'value', 0);
set(handles.radiobutton2, 'value', 0);
set(handles.text5, 'String', '');
set(handles.text8, 'string', '');
set(handles.text9, 'string', '');
set(handles.checkbox1, 'value', 0);
set(handles.checkbox2, 'value', 0);
set(handles.checkbox3, 'value', 0);
msgbox('Seleccione de nuevo todos los parámetros', 'Detección de
Imágenes ');

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
delete(handles.figure1)
EID_final

% --- Executes on button press in checkbox1.
function checkbox1_Callback(hObject, eventdata, handles)
% hObject    handle to checkbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject, 'Value') returns toggle state of checkbox1

% --- Executes on button press in checkbox2.
function checkbox2_Callback(hObject, eventdata, handles)

```

```
% hObject    handle to checkbox2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of checkbox2

% --- Executes on button press in checkbox3.
function checkbox3_Callback(hObject, eventdata, handles)
% hObject    handle to checkbox3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of checkbox3

% --- Executes on button press in radiobutton1.
function radiobutton1_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of radiobutton1
set(handles.radiobutton1,'value',1);
set(handles.radiobutton2,'value',0);

% --- Executes on button press in radiobutton2.
function radiobutton2_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of radiobutton2
set(handles.radiobutton1,'value',0);
set(handles.radiobutton2,'value',1);
```

Algoritmo para el análisis de una imagen en el modelo de color HSV

```

clc
clear all
close all
im=imread('imagenes 2/2010_0527FOTOSPROYECTOPIFI0135_1.jpg');
x=double(im(:,:,1));
y=double(im(:,:,2));
z=double(im(:,:,3));
imagen=rgb2hsv(im);
H=imagen(:,:,1);%H
S=imagen(:,:,2);%S
V=imagen(:,:,3);%V
[m n]=size(x);
%Matrices nuevas
x1=zeros(m,n);
y1=zeros(m,n);
z1=zeros(m,n);
%Eliminación de tonos que no son piel
for i=1:m
    for j=1:n
        if H(i,j)<=0.12 && S(i,j)>0.15 && S(i,j)<0.9 && V(i,j)>0.1
&& V(i,j)<0.95
            x1(i,j)=x(i,j);
            y1(i,j)=y(i,j);
            z1(i,j)=z(i,j);
        end
    end
end
%Creación de la nueva imagen
imp(:,:,1)=x1;
imp(:,:,2)=y1;
imp(:,:,3)=z1;
imp=uint8(imp);

%Calculo de porcentaje de pixeles de piel
cont=0;
for i=1:m
    for j=1:n
        if x1(i,j)~=0;
            cont=cont+1;
        end
    end
end
porcen_piel=(cont/(m*n))*100;
disp(porcen_piel)
figure
imshow(im)
figure
imshow(imagen)
figure
imshow(imp)

```

Algoritmo para el análisis de una imagen en el modelo de color YCbCr

```

clc
clear all
close all
im=imread('prueba.jpg');
x=im(:,:,1);
y=im(:,:,2);
z=im(:,:,3);
imagen=rgb2ycbcr(im);
Y=imagen(:,:,1);%Y
Cb=imagen(:,:,2);%Cb
Cr=imagen(:,:,3);%Cr
[m n]=size(x);
%Matrices nuevas
x1=zeros(m,n);
y1=zeros(m,n);
z1=zeros(m,n);
%Eliminación de tonos que no son piel
for i=1:m
    for j=1:n
        if Cb(i,j)>=80 && Cb(i,j)<=120 && Cr(i,j)>=133 &&
Cr(i,j)<=173
            x1(i,j)=x(i,j);
            y1(i,j)=y(i,j);
            z1(i,j)=z(i,j);
        end
    end
end
% %Calculo de porcentaje de pixeles de piel
cont=0;
for i=1:m
    for j=1:n
        if x1(i,j)~=0;
            cont=cont+1;
        end
    end
end
porcen_piel=(cont/(m*n))*100;
%Creación de la nueva imagen
imp(:,:,1)=x1;
imp(:,:,2)=y1;
imp(:,:,3)=z1;
disp('El porcentaje de piel es:');
disp(porcen_piel)
figure, imshow(im)
figure, imshow(imagen)
figure, imshow(uint8(imp))

```

Algoritmo para el análisis de un directorio en el modelo de color HSV

```

clc
clear all
close all
%seleccionamos el directorio
dir1=uigetdir();
%Seleccionamos sólo los archivos jpg
directorio=strcat(dir1,'/*.jpg');
%Cargamos los archivos
files=dir(directorio);
%Medimos la longitud de la variable para saber cuantas imágenes son
tam=length(files);
l=1;
for k=1:tam
    clear imp
    clear imp2
    close all
    nombreimagen=files(k,1).name;
    dir_imagen=strcat(dir1,'\',nombreimagen);
    im=imread(dir_imagen);
    x=im(:,:,1);
    y=im(:,:,2);
    z=im(:,:,3);
    imagen=rgb2hsv(im);
    H=imagen(:,:,1);%H
    S=imagen(:,:,2);%S
    V=imagen(:,:,3);%V
    [m n]=size(x);
    %Matrices nuevas
    x1=zeros(m,n);
    y1=zeros(m,n);
    z1=zeros(m,n);
    %Eliminación de tonos que no son piel
    for i=1:m
        for j=1:n
            if H(i,j)<0.25 && S(i,j)>0.15 && S(i,j)<0.9 &&
V(i,j)>0.15 && V(i,j)<0.95
                x1(i,j)=x(i,j);
                y1(i,j)=y(i,j);
                z1(i,j)=z(i,j);
            end
        end
    end
    imp(:,:,1)=x1;
    imp(:,:,2)=y1;
    imp(:,:,3)=z1;
    %Calculo de porcentaje de pixeles de piel
    cont=0;
    for i=1:m
        for j=1:n
            if x1(i,j)~=0;
                cont=cont+1;
            end
        end
    end
end

```

```
        end
    end
    porcen_piel=(cont/(m*n))*100;
    final{k,1}=nombreimagen;
    final{k,2}=porcen_piel;
    if porcen_piel>30
        res(1,1)=tam;
        res(1,2)=porcen_piel;
        l=l+1;
        nombre=strcat('positivas/',nombreimagen, '.jpg');
        imwrite(im,nombre);
    end
    fprintf('%d: %s      Porcentaje Piel %f
\n',k,nombreimagen,porcen_piel)
end
```


Algoritmo para el análisis de un directorio en el modelo de color YCbCr

```

clc
clear all
close all
%seleccionamos el directorio
dir1=uigetdir();
%Seleccionamos sólo los archivos jpg
directorio=strcat(dir1,'/*.jpg');
%Cargamos los archivos
files=dir(directorio);
%Medimos la longitud de la variable para saber cuantas imágenes son
tam=length(files);
l=1;
for k=1:tam
    clear imp
    clear imp2
    close all
    nombreimagen=files(k,1).name;
    dir_imagen=strcat(dir1,'\',nombreimagen);
    im=imread(dir_imagen);
    x=im(:,:,1);
    y=im(:,:,2);
    z=im(:,:,3);
    imagen=rgb2ycbcr(im);
    Y=imagen(:,:,1);%Y
    Cb=imagen(:,:,2);%Cb
    Cr=imagen(:,:,3);%Cr
    [m n]=size(x);
    %Matrices nuevas
    x1=zeros(m,n);
    y1=zeros(m,n);
    z1=zeros(m,n);
    %Eliminación de tonos que no son piel
    for i=1:m
        for j=1:n
            if Cb(i,j)>=80 && Cb(i,j)<=120 && Cr(i,j)>=133 &&
Cr(i,j)<=173
                x1(i,j)=x(i,j);
                y1(i,j)=y(i,j);
                z1(i,j)=z(i,j);
            end
        end
    end
    imp(:,:,1)=x1;
    imp(:,:,2)=y1;
    imp(:,:,3)=z1;
    %Calculo de porcentaje de pixeles de piel
    cont=0;
    for i=1:m
        for j=1:n
            if x1(i,j)~=0;
                cont=cont+1;
            end
        end
    end
end

```

```
        end
    end
    porcen_piel=(cont/(m*n))*100;
    final{k,1}=nombreimagen;
    final{k,2}=porcen_piel;
    if porcen_piel>45
        res(1,1)=tam;
        res(1,2)=porcen_piel;
        l=l+1;
        nombre=strcat('positivas/',nombreimagen);
        imwrite(im,nombre);
    end
    fprintf('%d: %s      Porcentaje Piel %f
\n',k,nombreimagen,porcen_piel)
end
```

Algoritmo para el análisis de una imagen en el modelo de color HSV usando la apertura morfológica

```

clc
clear all
close all
im=imread('imagenes/40006.jpg');
x=im(:,:,1);
y=im(:,:,2);
z=im(:,:,3);
imagen=rgb2hsv(im);
H=imagen(:,:,1);%H
S=imagen(:,:,2);%S
V=imagen(:,:,3);%V
[m n]=size(x);
%Matrices nuevas
x1=zeros(m,n);
r=zeros(m,n);
g=zeros(m,n);
b=zeros(m,n);
figure
imshow(imagen)

%Eliminación de tonos que no son piel
for i=1:m
    for j=1:n
        if H(i,j)<=0.12 && S(i,j)>0.15 && S(i,j)<0.9 &&
V(i,j)>0.1 && V(i,j)<0.95
            x1(i,j)=1;
        end
    end
end
se = strel('disk',5);
I_opened = imopen(x1,se);
P = bwlabeln(I_opened);
imagen1 = label2rgb(P);
figure
imshow(imagen1)

%Contar el número de pixeles de conectados por zonas
for t=1:max(max(P))
    cont=0;
    for i=1:m
        for j=1:n
            if P(i,j)==t
                cont=cont+1;
            end
        end
        contador(t)=cont;
    end
end
end

%    %Eliminar las zonas con menos de 5000 pixeles conectados

```

```

%
%   for t=1:max(max(P))
%       aux=contador(t)/max(contador);
%       if aux<0.1 && contador(t)~=0
%           I_opened(P == t)=0;
%           contador(t)=0;
%       end
%   end

for i=1:m
    for j=1:n
        if I_opened(i,j)==1
            r(i,j)=x(i,j);
            g(i,j)=y(i,j);
            b(i,j)=z(i,j);
        end
    end
end
%Creación de la nueva imagen
imp(:,:,1)=r;
imp(:,:,2)=g;
imp(:,:,3)=b;
imp=uint8(imp);

figure
imshow(imp)

if isempty(contador)
    pielfinal=0;
else
    pielfinal=max(contador)/sum(contador)*100
end

%   %Calculo de porcentaje de pixeles de piel
cont2=0;
for i=1:m
    for j=1:n
        if I_opened(i,j)~=0;
            cont2=cont2+1;
        end
    end
end
porcen_piel=(cont2/(m*n))*100

```

Algoritmo para el análisis de una imagen en el modelo de color YCbCr usando la apertura morfológica

```

clc
clear all
close all
contador=[];
im=imread('prueba 1/145.jpg');
x=double(im(:,:,1));
y=double(im(:,:,2));
z=double(im(:,:,3));
imagen=rgb2ycbcr(im);
Y=imagen(:,:,1);%H
Cb=imagen(:,:,2);%S
Cr=imagen(:,:,3);%V
[m n]=size(x);
%Matrices nuevas
x1=zeros(m,n);
r=zeros(m,n);
g=zeros(m,n);
b=zeros(m,n);
%Eliminación de tonos que no son piel
for i=1:m
    for j=1:n
        if Cb(i,j)>=80 && Cb(i,j)<=120 && Cr(i,j)>=133 &&
Cr(i,j)<=173
            x1(i,j)=1;
        end
    end
end
figure
imshow(im)
%Creación de la nueva imagen
figure
imshow(x1)
se = strel('disk',5);
I_opened = imopen(x1,se);
figure
imshow(I_opened,[])
P = bwlabeln(I_opened);
imagenl = label2rgb(P);
figure
imshow(imagenl)
%Contar el número de pixeles de conectados por zonas
for t=1:max(max(P))
    cont=0;
    for i=1:m
        for j=1:n
            if P(i,j)==t
                cont=cont+1;
            end
        end
        contador(t)=cont;
    end
end
end

```

```

end

total=sum(contador);
porcentajepiel=max(contador)/total*100

for t=1:max(max(P))
    aux=contador(t)/max(contador);
    if aux<0.1 && contador(t)~=0
        I_opened(P == t)=0;
        contador(t)=0;
    end
end

if isempty(contador)
    pielfinal=0
else
    pielfinal=max(contador)/sum(contador)*100
end

%     pielfinal=sum(contador)/total * 100
for i=1:m
    for j=1:n
        if I_opened(i,j)==1
            r(i,j)=x(i,j);
            g(i,j)=y(i,j);
            b(i,j)=z(i,j);
        end
    end
end
imp(:,:,1)=r;
imp(:,:,2)=g;
imp(:,:,3)=b;
% imrec=ycbcr2rgb(imp);
%Calculo de porcentaje de pixeles de piel
cont=0;
for i=1:m
    for j=1:n
        if r(i,j)~=0;
            cont=cont+1;
        end
    end
end

porcen_piel=(cont/(m*n))*100

figure
imshow(uint8(imp))

```

ANEXO B: PUBLICACIONES

Detección de imágenes con contenido explícito

Jorge Alberto Marcial Basilio^{#1}, Gualberto Aguilar Torres^{#2}, Gabriel Sánchez Pérez^{#3}

Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacan
Avenida Santa Ana #1000 Col. San Francisco Culhuacan, Deleg. Coyoacán C.P. 04430, México D.F.

¹ jmarcialb85@gmail.com

² autg79y@yahoo.com

³ caaann@gmail.com

Resumen—En este artículo se presenta un sistema para la detección de imágenes con contenido explícito basado en algoritmos de Visión por Computadora, Reconocimiento de Patrones y el software FTK Explicit Image Detection.

Primero se utiliza un modelo de color diferente al utilizado en las imágenes de entrada, con la finalidad de discriminar los elementos de la imagen que no sean piel humana dentro de la imagen. Después se filtran estas imágenes usando la detección de piel. La imagen de salida es una imagen que contiene solo las zonas de piel con las que está compuesta. Los resultados obtenidos muestran una comparación entre el sistema propuesto y el software de la empresa Access Data llamado Forensic Toolkit 3.1.

Keywords—Detección de piel, HSV, con contenido explícito, Reconocimiento de Patrones.

I. Introducción

Con el desarrollo de Internet, la dramática caída de los costos de almacenamiento de datos y el avance en la tecnología de codificación, se está generando una impresionante variedad de fotografías, animaciones, gráficos, sonidos y videos. En la actualidad es fácil tener acceso a una computadora con conexión a Internet, en donde existe una gran cantidad de imágenes para adultos que se pueden descargar de forma gratuita. Este tipo de medios está al alcance de los niños, por lo que muchos padres están preocupados por este problema creciente [1].

Filtrar las imágenes con contenido clasificado como para adultos es muy importante para los motores de búsquedas de los principales programas de navegación en Internet con la finalidad de evitar contenido ofensivo. Actualmente, existen algunas formas de detener las imágenes pornográficas, que llegan a las computadoras, así como el bloqueo de sitios no deseados o la identificación de imágenes que muestran contenido explícito. Existen algunos programas en el mercado que nos permiten bloquear sitios con contenido ofensivo o explícito como: CyberPatrol, NetNanny, Family.net y K9 Web Protection, etc. Los cuales proveen a los padres un control para salvaguardar a sus hijos que navegan por Internet.

Algunos otros programas que existen son los que detectan las imágenes pornográficas existentes en la computadora, como el que ofrece la empresa SurfRecon; existe otro software que a pesar de tratarse de una herramienta de informática forense, también ayuda en la detección del contenido explícito dentro de las imágenes, el nombre de esta herramienta es FTK Explicit Image Detection, la cual se encuentra a partir de la actual versión de Forensic Toolkit 3.1 (FTK 3.1). Otra forma de identificar imágenes pornográficas se lleva a cabo por medio de análisis de texto o visión por computadora. Otros trabajos realizados en este tema son: el desarrollado por Forsyth y Fleck, quienes diseñaron un software para detectar este tipo de imágenes [2] y el de Wiederhold y Wang, que diseñaron un algoritmo para el reconocimiento de imágenes con contenido censurable [3] y Li Chen *et al.* Diseñaron un detector de piel basado en Redes Neuronales [4].

En este artículo se presenta un nuevo algoritmo de detección de imágenes pornográficas. Este algoritmo está basado en algoritmos de visión por computadora y el reconocimiento de patrones. Las imágenes se filtraron primeramente usando un modelo de color distinto al de las imágenes de entrada, y después se utilizó un clasificador para saber si las imágenes tienen algún contenido explícito.

En la sección 2 se presenta una breve descripción del sistema propuesto, una introducción del modelo de color HSV, y la representación del detector de piel utilizado para este sistema se presentan en la sección 3 y 4 respectivamente, en la sección 5 resultados y por ultimo conclusiones.

II. Sistema propuesto

En la Figura 1 se muestra el sistema propuesto, el cual consta de dos partes: detector de textura y un clasificador.

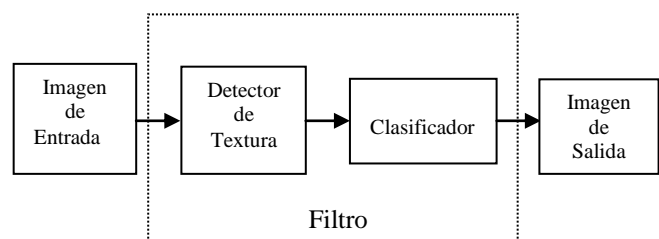


Fig. 1 Sistema Propuesto

ROC&C'2010 - CP-14 PONENCIA RECOMENDADA
POR EL **COMITÉ DE COMPUTACION**
DEL **IEEE SECCIÓN MÉXICO** Y PRESENTADA EN LA
REUNIÓN DE OTOÑO, ROC&C'2010, ACAPULCO, GRO.,
DEL 28 DE NOVIEMBRE AL 4 DE DICIEMBRE DEL 2010.

La primera parte consta de un detector de textura, el cual se utilizó, usando un modelo de color, que nos permite discriminar algunas partes de la imagen de entrada que no necesitamos de una manera eficaz, esto se logró implementando el modelo de color HSV.

La segunda parte del sistema propuesto es crear un clasificador, usando el modelo de color HSV, que nos permite reconocer a las texturas dentro de la imagen, y así poder a través de la detección de piel saber si existen zonas de pixeles con algún tono de piel, y de esta manera saber su probabilidad de que fuera una imagen con contenido explícito. La detección de piel puede ser usada como la base para la detección de imágenes con contenido explícito, porque existe una importante relación entre las imágenes con un gran contenido de zonas con piel y las imágenes pornográficas ó con contenido explícito.

III. Modelos de color

3.1 Modelo de Color RGB

El modelo de color RGB (Red, Green, Blue) por sus siglas en Inglés, es un modelo de color aditivo que hace referencia a la composición del color en términos de la intensidad de los colores primarios: el rojo, el verde y el azul.

3.2 Modelo de Color HSV

El modelo de color HSV (Hue, Saturation, Value) por sus siglas en Inglés se trata de una transformación no lineal del espacio de color RGB, y los colores son una combinación de tres valores: el tono (H), la saturación o cantidad de color (S) y valor del mismo (V). Estos valores suelen representarse en un diagrama circular, como se muestra en la Figura 2.

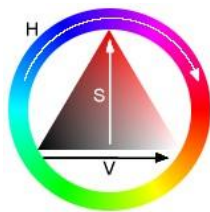


Fig. 2 Representación del modelo de color HSV

Las tres magnitudes pueden tener los siguientes valores:

- Tonalidad, el tipo de color (como por ejemplo: rojo, verde o amarillo). Se representan como un grado de ángulo cuyos valores posibles van de 0 a 360° (aunque para algunas aplicaciones se normalizan del 0 al 100%).
- Saturación. Se representa como la distancia al eje del brillo negro-blanco. Los valores posibles van del 0 al 100%.

- Valor del color. Representa la altura en el eje blanco-negro. Los valores posibles van del 0 al 100%. 0 siempre es negro. Dependiendo de la saturación, 100 podría ser blanco o un color más o menos saturado.

Haciendo uso de este modelo de color en la imagen de entrada, se convierte utilizando las expresiones matemáticas (1) al (3) mostradas a continuación.

$$H = \arccos \frac{\frac{I}{2} [(R-G) + (R-B)]}{\sqrt{[(R-G)^2 + (R-B)(G-B)]}} \quad (1)$$

$$S = 1 - 3 \frac{\min(R, G, B)}{R + G + B} \quad (2)$$

$$V = \frac{I}{3} (R + G + B) \quad (3)$$

Una vez hecha la transformación de la imagen de entrada, se observó que los tonos de piel de una persona se podían ver en diferente color al de los que se pueden ver en los diferentes objetos que están dentro de la misma imagen, esto se muestra en la Figura 3, la cual es una muestra de la conversión al modelo HSV de una imagen.



Fig. 3 Conversión de la imagen

(a) Imagen en RGB (b) Imagen en HSV

Como se mencionó una ventaja al utilizar este modelo de color en particular, es que podemos descartar muchas de los objetos utilizando un filtro, en este caso utilizaremos la detección de piel con el fin de solo quedarnos con las zonas de la piel, que son las más importantes para nuestro propósito.

IV. Detección de piel

La detección de piel es un indicador de la presencia de una extremidad, torso o rostro dentro de una imagen digital. En los últimos años varios métodos de identificación de piel dentro de una imagen han sido desarrollados. El color de piel ha resultado ser una forma útil y robusta para la detección, localización y seguimiento del rostro. Investigadores han analizado el uso de la información del color para detectar la piel. Jones y Regh [5] construyeron un modelo de histograma de color usando técnicas de aprendizaje en el espacio de color RGB. Yang y Auhuja [6] estimaron la función de densidad de probabilidad del color de la piel a partir del modelo finito de

mezcla Gaussiana, cuyos parámetros se estiman mediante el algoritmo EM. Existen otras investigaciones que también tratan los diferentes modelos de detección de piel que existen como Vezhnevets *et al.*[7], Kelly *et al.*[8].

En este artículo se propone una nueva solución utilizando el modelo de color mencionado en la sección anterior, el cual es muy similar al modelo de color RGB.

Una vez hecho el cambio de modelo de color, se procede a la detección de píxeles con color de piel, esto se logró observando varias imágenes, de las cuales se obtuvo un umbral en donde la mayoría de las personas con diferente color de piel dentro de la imagen puede ser segmentada.

Para determinar el umbral fue necesario el análisis de los histogramas en el modelo de color HSV, como se observa en la Figura 4, para mayor facilidad se tomó una imagen de un rostro para poder identificar con mayor claridad el umbral que necesitaríamos utilizar.

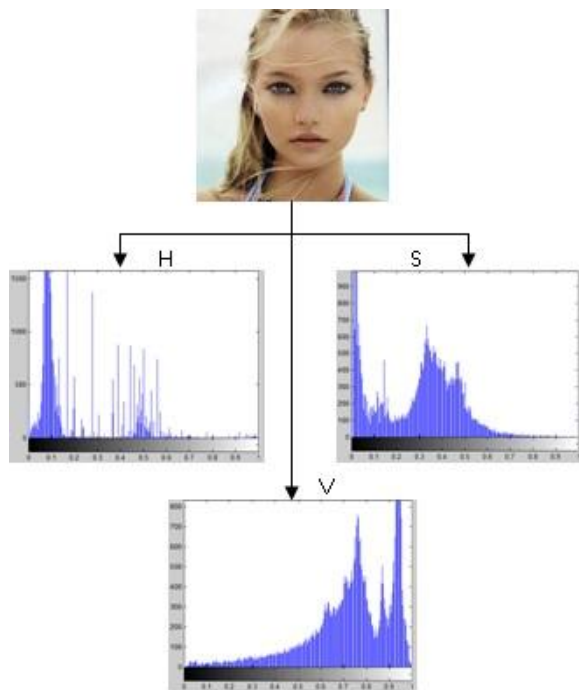


Fig. 4 Histogramas del modelo de color HSV

Los histogramas observados en la Figura 4 nos ayudan a tomar una idea de los valores con el cual se podría tomar para elegir un umbral, que fuera capaz de tomar los valores de la piel, esto sería correcto si solo se deseara detectar personas con el mismo color de piel de la persona que se usó como referencia, pero debido a que en Internet existe una gran variedad de imágenes, que no solo contienen a personas de un color en específico, también hay personas con distinto color de piel, por lo tanto después de un análisis más profundo, el umbral que se decidió utilizar fue el siguiente:

$$H < 0.00 \text{ y } H < 0.25$$

$$S > 0.15 \text{ y } S < 0.90$$

$$V > 0.20 \text{ y } V < 0.95$$

Donde H, S, V están dentro del rango de 0 a 1.

Con la finalidad de encontrar personas desnudas, también se utiliza otro tipo de características, como el porcentaje de píxeles detectados como piel. En base a estas características, un procedimiento de segmentación se realiza en imágenes a color. Algunos ejemplos son ilustrados en la Figura 5, la cual muestra a personas con diferentes colores de piel, y se puede observar que el umbral utilizado funciona de forma adecuada, ya que de una imagen solo toma las zonas de piel existentes dentro de ella.

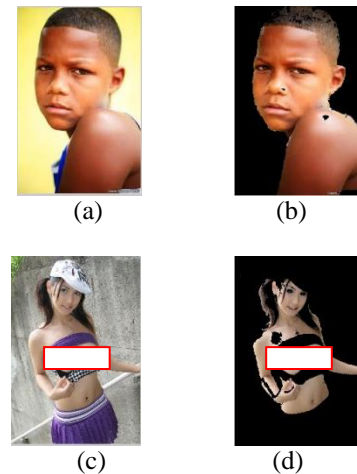


Fig. 5 Segmentación por color de piel

(a) y (c) Imágenes Originales (b) y (d) Imágenes Segmentadas

El umbral propuesto en esta sección fue capaz de detectar zonas de color de piel de manera eficaz.

El método propuesto para la detección de piel puede encontrar regiones de piel, pero para determinar cuáles de un conjunto amplio de imágenes contienen desnudos es aún un gran desafío.

V. Desarrollo

Para comprobar el funcionamiento del algoritmo se realizaron pruebas usando diferentes imágenes, las cuales se obtuvieron de Internet. El sistema puede tomar diferentes tipos de imágenes, como imágenes en distintas condiciones de iluminación y de distinto tamaño.

Las imágenes de prueba se clasificaron en: a) imágenes de personas desnudas o con contenido explícito, y b) imágenes naturales. Dentro de las imágenes con contenido explícito se pueden encontrar personas Asiáticas, Caucásicas, Europeas, Latinas y una pequeña cantidad de personas de piel negra. En las imágenes naturales que se recolectaron de Internet

podemos encontrar imágenes de personas vestidas, animales, plantas, automóviles, dibujos animados, paisajes y entre otras.


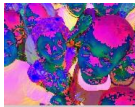


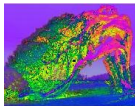







Primeramente la imagen de entrada se convierte del modelo de color RGB al modelo HSV para que de esta esa forma se descarten los objetos que no interesan de la imagen original, después se usa la detección de piel, para que se pueda identificar las zonas que contengan algún color de piel, y así solamente quedarse con la imagen de la persona o las personas que estén en la imagen de entrada.

Una vez que se ha llegado a este punto se toman todas las zonas de piel detectadas, se procede a contar la cantidad de pixeles que existen dentro de la imagen para conocer la probabilidad de que esta imagen sea catalogada como imagen con contenido explícito o no, los resultados obtenidos se muestran en la tabla I.

Para poder determinar si la imagen tiene contenido explícito, se utilizó la expresión matemática descrita en (4), y así obtener el porcentaje de la cantidad de piel existente en la imagen, si el porcentaje es mayor al 50%, se considera que tiene contenido censurable.

$$\text{porcentaje piel} = \frac{\#total \text{ pixeles color piel}}{\#total \text{ pixeles de la imagen}} \times 100 \quad (4)$$

Tabla I
Resultados de la Clasificación usando la Metodología Propuesta





Imagen de Entrada	Clasificación de imágenes		
	Imagen en HSV	Clasificador Propuesto	Cantidad de pixeles del color de la piel
			27.74 %
			26.10%
			58.17%
			64.78%

Este sistema sirve de referencia para saber si las imágenes analizadas cuentan con cierta cantidad de pixeles con color de piel, este es un factor para determinar si las imágenes contienen desnudos, ya que la mayoría de las imágenes de desnudos se componen de zonas de piel que ocupan la mayor parte de la imagen.

Se puede observar que el umbral seleccionado para la segmentación de las imágenes funciona de mejor manera en personas de piel clara, aunque para personas morenas o personas de color negro, este umbral no segmenta el total de las zonas de la piel, sin embargo, no se pierden detalles en la detección de dichas personas, aun así se puede estimar su probabilidad de que la imagen contenga desnudos.

También se utilizó el software de la empresa Access Data, Forensic Toolkit 3.1, para hacer la comparación entre el sistema aquí propuesto y su sistema de detección de imágenes, este software utiliza tres algoritmos para saber el porcentaje del contenido explícito de las imágenes, para realizar estas pruebas se usaron las mismas imágenes de entrada. En la tabla II se muestran los resultados obtenidos con dicho programa.

Tabla II
Resultados obtenidos con FTK 3.1

Imagen de Entrada	Algoritmos utilizados		
	X-FST	X-DFT	X-ZFN
	99%	0%	99%
	99%	21%	99%
	99%	99%	99%
	0%	0%	0%

El software de Access Data utilizado solo sirve para ayudar al usuario en la toma de decisiones, en cuanto si las imágenes procesadas son o no son imágenes con algún contenido explícito, como podemos observar en la tabla II, existe en algunas imágenes una discrepancia entre el porcentaje que entrega como resultado y el contenido de la imagen original.

Conclusiones

En este trabajo se presenta una solución para la detección de imágenes con contenido explícito en imágenes de color, usando el modelo de color HSV y un modelo para la detección de piel, el cual funciona eficazmente, aunque en algunas imágenes puede presentar errores esto debido a las condiciones de iluminación en las que haya sido tomada la imagen, otro factor que puede causar una mala interpretación del sistema son las imágenes que han sido modificadas o retocadas utilizando algún software usado para este propósito.

El sistema propuesto entrega una imagen que solo muestra a la persona o personas que estén en la imagen, así como un

porcentaje de píxeles que tienen color de piel, este factor sirve como un factor para la saber si la imagen es o no para adultos.

Aun falta trabajo para saber de forma precisa si las imágenes que se encuentran en Internet contienen personas desnudas, y así poder implementar un sistema que no nos permita ver este tipo de imágenes.

REFERENCIAS

- [1] L. Duan, G. Cui, W. Gao and H. Zhang, "Adult Image Detection Method Base-on Skin Color Model and Support Vector Machine", The 5th Asian Conference on Computer Vision, 23-25 January 2002, Melbourne, Australia.
- [2] D. A. Forsyth, M. Fleek, and C. Bregler, "Finding naked people", Proc. Forth European Conference on Computer Vision. pp 593-602, 1996.
- [3] G. Wiederhold and J. Z. Wang, "WIPE (TM): Wavelet Image Pornography Elimination; A System for Screening Objectionable Images".
- [4] Li Chen, J. Zhou, Z. Liu, W. Chen and G. Xiong, "A Skin Detector Based on Neural Network", In IEEE 2002 International Conference on Communication, Circuits and Systems and West Sino Expositions, vol. 1, 615 – 619, 2002.
- [5] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection", In Proc. Of the CVPR '99, vol. 1, 274-280, 1999
- [6] M. H. Yang and N. Ahuja, "Detecting human faces in color images", In International Conference on Image Processing (ICIP), vol 1, 127-130, 1998.
- [7] V. Vezhnevets, V. Sazonov, A. Andreeva, "A Survey on Pixel-based Skin Color Detection Techniques", In Proceedings of the GrapiCon, 85-92, 2003.
- [8] W. Kelly, A. Donnellan and D. Molloy, "Screening for Objectionable Images: A Review of Skin Detection Techniques", In Machine Vision and Image Processing Conference (IMVIP '08), 151 – 158, 2008.

Biografía



Jorge Alberto Marcial Basilio recibió el título de Ingeniero en Comunicaciones y Electrónica en 2009, en el Instituto Politécnico Nacional. Sus principales áreas de interés son procesamiento digital de imágenes, reconocimientos de patrones, redes neuronales, biometría. Actualmente se encuentra estudiando la Maestría en Ingeniería en Sistemas y Tecnologías de la Información en el Instituto Politécnico Nacional.



Gualberto Aguilar Torres recibió el grado de Maestro en Ciencias de Ingeniería en Microelectrónica y el grado de Doctor en Comunicaciones y Electrónica en 2004 y 2008 respectivamente, en el Instituto Politécnico Nacional. Sus principales áreas de interés son procesamiento de señales, reconocimiento de patrones, redes neuronales, biometría y campos relacionados. En 2005 recibió el premio a la mejor tesis de maestría por parte del IPN. El profesor Aguilar Torres es miembro del SNI.



Gabriel Sánchez Pérez recibió el grado de Ingeniero en Computación y el grado de Doctor en Comunicaciones y Electrónica en 1999 y 2005 respectivamente, en el Instituto Politécnico Nacional.

De Enero 2001 a Octubre 2006 él se afilió como profesor asistente al Departamento Ingeniería en Computación de la Escuela Superior de Ingeniería Mecánica y Eléctrica del Instituto Politécnico Nacional. En Octubre 2006 se incorporó a la Sección de Estudios de Posgrado e Investigación del Instituto

Politécnico Nacional donde actualmente es profesor. El profesor Sánchez Pérez es miembro de IEEE y miembro del Sistema Nacional de Investigadores de México.

EXPLICIT CONTENT IMAGE DETECTION

Jorge Alberto Marcial Basilio¹, Gualberto Aguilar Torres¹, Gabriel Sánchez Pérez¹, Linda Karina Toscano Medina¹, Héctor Manuel Pérez Meana¹, Enrique Escamilla Hernandez¹

¹Graduate School, Instituto Politécnico Nacional ESIME Culhuacan
Avenida Santa Ana #1000 Col. San Francisco Culhuacan, Deleg. Coyoacán C.P. 04430,
México D.F.

¹{jmarcialb0300, gaguilar, gsanchezp, ltoscano, hmperezm, eescamillah}
@ipn.mx

ABSTRACT

This paper proposes a system gives for explicit content image detection based on Computer Vision Algorithms, pattern recognition and FTK software Explicit Image Detection. In the first stage, HSV color model is used for the input images for the purpose of discriminating elements that are not human skin images. Then the image is filtered using skin detection. The output image only contains the areas of which it is composed. The results show a comparison between the proposed system and the company software Access Data called Forensic Toolkit 3.1 Explicit Image Detection is performed.

KEYWORDS

Skin Detection, HSV Color Model, Explicit Content, Pattern Recognition, Computer Vision

1. INTRODUCTION

With the development of Internet, dramatically falling costs of data storage and advances in coding technology are generating a dazzling array of photography, animation, graphics sound and video [1]. Nowadays it is easy to have access to a computer with an Internet connection where there is currently a large amount of adult images for free downloading. This kind of media is also available for children and is an increasingly problem for many parents.

Filtering images with adult classified content is very important for searching principal Internet browser programs to avoid offensive content. Nowadays there are some ways to stop pornographic images on computers, such as blocking unwanted sites or identifying images that show explicit content. There are some programs in the foreign market that allow blocking sites on Internet with offensive or explicit content such as: CyberPatrol, ContentProtect, NetNanny, Family.net and K9 Web Protection [2]. All these programs provide parental control to safeguard their children using the Internet. There are some others programs which detect pornographic images within the computer such as: SurfRecon that offers a program for this purpose, and despite being a tool of computer forensic, helps to detect images with explicit content.

The name of this tool is FTK Explicit Image Detection, which comes in the "FTK 3.1" version. There are some papers on this subject such as: the paper carried out by Forsyth and Fleck who designed software to detect naked people [3], Wiederhold and Wang design an algorithm for recognition of images with doubtful content [4], and Li Chen *et al* design a skin detector based-on Neural Network [5].

There are some investigators who carried out papers about adult image detection as: Xiaoyin Wang *et al.* [6] who proposed an algorithm to detect adult images, Yue Wang *et al.* [7] who proposed a way to help the algorithms to detect objectionable images using nipple detection, Huicheng Zheng *et al.* [8] designed a filtering system to adult images, Wonil Kim *et al.* [9] design

a neural network based adult image classification, Jiann-Shu Lee *et al.*[10]proposed an algorithm to naked image detection based on adaptive and extensible skin color model.

In this paper a new algorithm to detect explicit images is proposed. It is based on Computer Vision algorithms and pattern recognition techniques. First the images are changed from the color model to discriminate objects in the image of no interest. In the next part of the proposed system the image is filtered using skin detection, with the aim to segment a person or people within the image. Then we can estimate the probability of the image as an image with explicit content, by counting all pixels with some skin tone.

The paper is organized as follows. An introduction of color models RGB and HSV, and the representation of skin detector used for the system are presented in the section 3 and 4 respectively. Section 4 presents a brief description of the proposed system, and in section 5 are the results. Finally the conclusions are given.

2. COLOR MODELS

2.1. RGB Color Model

The RGB color model is an additive color model in which the primary colors red, green, and blue light are added together in various ways to reproduce a broad array of colors. The name comes from the initials of the three colors Red, Green, and Blue. The RGB color model is shown in the Figure 1.

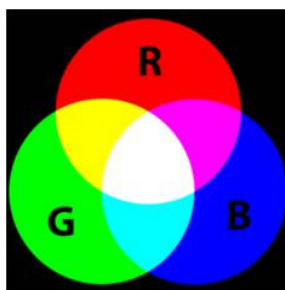


Figure 1. RGB Color Model

The main purpose of the RGB color model is for sensing, representation, and display of images in electronic systems, such as televisions and computers.

The RGB color model is an additive in the sense that three light beams are added together to make a final color. To form a color with RGB, three colored light beams (one red, one green, and one blue) should be superimposed. Each of the three beams is called a component of that color, and each can have arbitrary intensity, from fully off to fully on, in the mixture. Zero intensity for each component gives the darkest color (no light, considered the black), and full intensity of each gives a white.

A color in the RGB color model is described by indicating how much of each of the red, green, and blue is included in each component which can vary from zero to a defined maximum value which depends of the application. In computing, the component values are often stored as integer numbers in the range 0 to 255.

2.2. HSV Color Model

HSV color model (Hue, Saturation, and Value) is a non-linear transformation of the RGB space color, and the colors are a combination of the three values: the Hue (H), Saturation or color quantity (S), and itself value (V). These values are represented in a circular diagram, as shown in Figure 2.

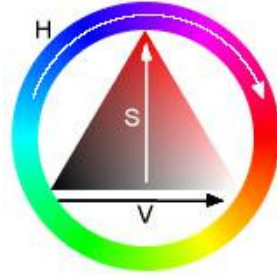


Figure 2. HSV Color Model

The three magnitudes can have the following values

- Hue: The type of color (e.g. red, green, or yellow). These are represented as a degree of angle whose possible values range from 0 to 360° (although for some applications are normalized from 0 to 100%).
- Saturation: Is represented as the distance from the axis of the black-white glow. The possible values range from 0 to 100%.
- Value: Represents the height in the black-white axis. The possible values range from 0 to 100%. 0 is always black. Depending on the saturation, 100 could be white or a more or less saturated color.

Using this color model as an input image is converted using the mathematical expressions (1) to (3) that are shown below.

$$H = \arccos \frac{\frac{1}{2} [(R-G) + (R-B)]}{\sqrt{[(R-G)^2 + (R-B)(G-B)]}} \quad (1)$$

$$S = 1 - 3 \frac{\min(R,G,B)}{R+G+B} \quad (2)$$

$$V = \frac{1}{3} (R+G+B) \quad (3)$$

Once the transformation of the input image was made, it was observed that the skin tone of a person could be seen in a different color from those seen from different objects within the same image. An example of this is shown in Figure 3, which is a sample of conversion to the HSV color model of an image in RGB color model.

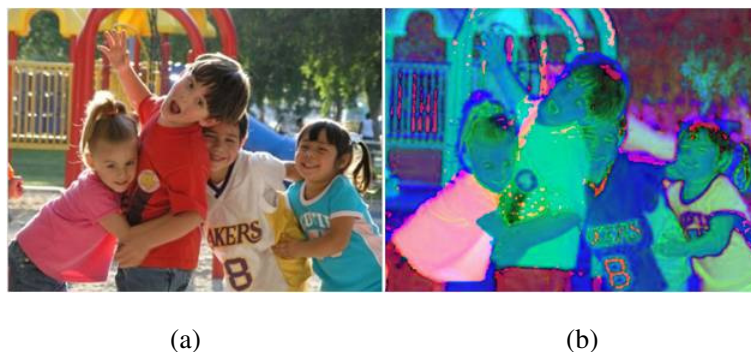


Figure 3. (a) RGB Image (b) HSV Image

As mentioned, an advantage of using this particular color model is that we can rule out many objects using a simple filter, in this case we use skin detection to get only the areas of skin which are the most important for our objective.

3. SKIN DETECTION

Skin detection can help detect a human limb, torso, or face within a picture. Lately many methods of skin identification within a digital image have been developed. Skin color has proved to be a useful and robust method for face detection, localization and tracking. There have been a number of researchers who have looked at using color information to detect skin. Jones and Rehg [11] constructed a color model using histogram-learning techniques at RGBcolor space. Yang and Auhuja [12] estimated probability density function of human skin color using a finite Gaussian mixture model whose parameters are estimated through the EM algorithm. There are other researchers who have developed papers about the different models of skin detection as Vezhnevets *et al.* [13], Kakumanu *et al.*[14], Kelly *et al.* [15].

In this paper a novel solution using the HSV color model, which is very similar to the RGB color model, is proposed.

Once the change of color model has been made, the next stage is to proceed to pixel detection with human skin. This was achieved by observing several images, which are a threshold where most people with different skin color within the image can be segmented.

To determine the threshold it was necessary to make an analysis of the histograms in the HSV color model. As observed in Figure 4 an image of the face of a girl identified with major clarity the threshold that we need.

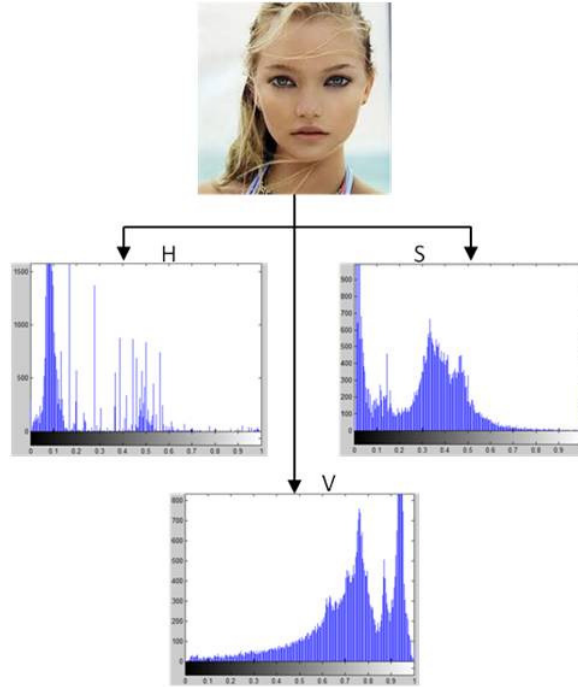


Figure 4. Histograms of HSV Color Model

The histograms observed in Figure 4 helps to have an idea of the values, which could be taken to choose a threshold able to take the skin values. This would be correct if only detecting people with the same skin color of the girl used as reference is desired but in Internet there exists a large amount of images that not only contain people with a specific skin color, but also people with different skin color, so after exhaustive analysis the threshold decided was the following:

$$H > 0 \text{ and } H < 0.25$$

$$S > 0.15 \text{ and } S < 0.9$$

$$V > 0.2 \text{ and } V < 0.95$$

Where H,S,V are in the range from 0 to 1.

With the purpose of finding naked people, there are other kinds of features such as the percentage of pixels detecting similar skin color. Based on these features, a procedure of segmentation is carried out in color images. Some examples are illustrated in Figure 5 which shows people with different skin color and can be seen that the threshold used works appropriately.



(a)

(b)

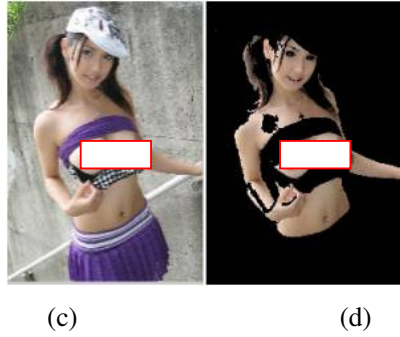


Figure 5. Skin Color Segmentation

(a) and (c) Original Images, (b) and (d) Segmented Images

The proposed threshold in this section was able to detect skin color zones effectively.

The method proposed for skin detection can find skin areas, but to decide which of a comprehensive set of images contain naked people is still a great challenge.

4. PROPOSED SYSTEM

In the Figure 6 shows the proposed system, which has one part: a classifier.

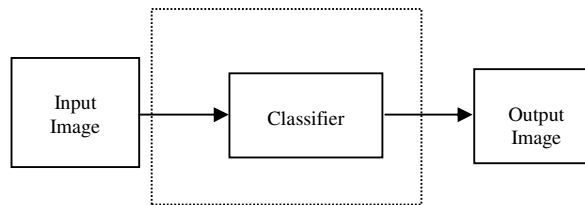


Figure 6. Proposed System

The proposed system is created a classifier, using the HSV color model, which allows us to recognize texture within the image, and this way can detect skin to know if there are pixel areas with some skin tone, and the value of the likelihood that the input image was an image with explicit content can be estimated.

Skin detection can be used as the basis for detection of the images with explicit content because there is a considerable relationship between the images with large areas of skin and pornographic images or with explicit content.

5. RESULTS

A test to prove the performance of the algorithm was used using different images from Internet. The system can process different kind of images, as images in different lighting conditions and images with different size.

The input images for testing are classified in: a) images of naked people or with explicit content and b) natural images. In the images with explicit content people Asians, Caucasian, Europeans, Latin Americans and a little amount of people with black skin can be found. In natural images there are different kinds of images such as: dressed people, animals, plants, cars, cartoons, landscapes and others were also obtained from Internet.





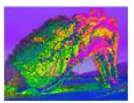







First the input image is transformed from the RGB to the HSV color model. Next stage using the threshold that is proposed in section 3 the image is filtered using the skin detection to identify the areas that contain some skin color, and in this way only get the image of the person or people within the input image.

At this point, all skin areas detected are taken, and proceed to count the amount of pixels that there are within the image to estimate the likelihood that this image is classified as an image with explicit content or not. The results obtained are shown in Table 1.

To be able determine if the input image has explicit content, a mathematic expression is proposed, described in (4), and in this way obtain the percentage of amount of skin that there are in the input image. If the percentage is more than 50% it is considered to have objectionable content.

$$skin\ percentage = \frac{\#skin\ color\ pixels}{\#image\ pixels\ in\ total} \times 100 \quad (4)$$

Table 1. Classification results using the proposed system

Input Image	Images Classification		
	HSV Image	Proposed Classifier	Skin Percentage
			27.74 %
			26.10%
			58.17%
			64.78%

The Table 1 shows four images of the set, which the proposed system was proved, in this results can be observed that the third and the fourth image are classified as an image with explicit content.

This system is used as reference to know if the analyzed images have a certain quantity of pixels with skin color. This is a factor to determine if the image has naked people, as the majority of images that have undressed people are made up of skin zones that take up most of the image.

Within the results, it can be observed that the selected threshold for the image segmentation works efficiently in people with white skin, although to brown people or black people, this threshold does not segment the total of skin areas. However, we can estimate its naked people likelihood with the fact that details are not lost detecting such things.

Also software was used from Access Data called Forensic Toolkit 3.1, to make the comparison between the proposed system and its system to detect images. It makes a forensic image of the device that have to analyze, after the analysis began with three algorithms to determine the percentage of explicit content of an image. The description of the three algorithms used for evaluate the images was obtained from [16] and was shown in the Table 2:

Table 2. Explicit Image Detection Profile Types

Profile Name	Level	Description
X- DFT	Default (XS1)	This is the most generally accurate, It is always selected.
X-FST	Fast (XTB)	This is the fastest. It scores a folder by the number of files it contains that meets the criteria for a high likelihood of explicit material. It is built on a different technology than X-DFT and does not use “regular” DNAs. It is designed for very high volumes, or real-time page scoring. Its purpose is to quickly reduce, or filter, the volume of data to a meaningful set.
X-ZFN	Less False Negatives (XT2)	This is a profile similar to S-FST but with more features and with fewer false negatives than X-DFT. Apply this filter after initial processing to all evidence, or to only the folders that score highly using the X-FST option. Check-mark or highlight those folders to isolate them for Additional Analysis.

To test its system, the same set of input images was used using the three profiles that have. The results of the software are shown in the Table 3.

Table 3. Results obtained with FTK 3.1

Input Image	Algorithms used		
	X-FST	X-DFT	X-ZFN
	99%	0%	99%
	99%	21%	99%
	99%	99%	99%
	0%	0%	0%

The software from Access Data is used only to help the users take decisions whether the images have or lack explicit content, as can be observed in the Table 3. There is a discrepancy between the percentage that gives the result and the content of original image.

6. CONCLUSIONS

This paper proposed an algorithm to detect images with explicit content in color images, using the HSV color model and a method of skin detection which works effectively although in some images it could find some errors, due to the image lighting conditions when taken, another factor that can be by a bad interpretation of the system.

HSV color model is an important method to be able to decrease all the lighting problems that the image could be had; moreover, using this color model is more visible the skin tone than the RGB color model, for this reason is used HSV color model to be able to do skin detection.

The proposed system gives an output image that only shows color skin pixels within the image, in basis to this can be known the likelihood that the image is an explicit content image or not, due to explicit content image in most part has color skin pixels.

The importance of the comparison between the proposed system and the software Forensic Toolkit 3.1 was done to know if the proposed system could do the same work, and this way know whether the input image is an explicit content image or not, at final could prove that the system proposed carry out effectively.

REFERENCES

- [1] L. Duan, G. Cui, W. Gao and H. Zhang, "Adult Image Detection Method Base-on Skin Color Model and Support Vector Machine", The 5th Asian Conference on Computer Vision, 23-25 January 2002, Melbourne, Australia.
- [2] C. Habis, and F. Krsmanovic, "Explicit Image Filter," Stanford Univ., 2005.
- [3] D. A. Forsyth, M. Fleek, and C. Bregler, "Finding naked people", Proc.Forth European Conference on Computer Vision. pp 593-602, 1996.
- [4] G. Wiederhold and J. Z. Wang, "WIPE (TM): Wavelet Image Pornography Elimination; A System for Screening Objectionable Images".
- [5] Li Chen, J. Zhou, Z. Liu, W. Chen and G. Xiong, "A Skin Detector Based on Neural Network", In IEEE 2002 International Conference on Communication, Circuits and Systems and West Sino Expositions, vol. 1, 615 – 619, 2002.
- [6] Xiaoyin Wang, Changzhen Hu, and Shuping Yao, "An Adult Image Recognizing Algorithm Based on Naked Body Detection", In Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on, 197 – 200, 2009.
- [7] Yue Wang, Jun Li, HeeLin Wang, and ZuJun Hou, "Automatic Nipple Detection Using Shape and Statistical Skin Color Information", Advances In Multimedia Modeling, LNCS, Vol. 5916/2010, 644 – 649, 2010.
- [8] Huicheng Zheng, Hongmei Liu, and Daoudi, M, "Blocking Objectionable Images: Adult Images and Harmful Symbols", In. Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on, Vol. 2, 1223 – 1226, 2004.
- [9] Wonil Kim, Han-Ku Lee, Seong Joon Yoo and Sung Wook Baik, "Neural NetworkBased Adult Image Classification",Artificial Neural Networks: Biological Inspirations – ICANN 2005 LNCS, Volume 3696/2005, 481 – 486, 2005.
- [10] Jiann-Shu Lee, Yung-Ming Kuo, Pau-Choo Chung, and E-Liang Chen, "An adult image Identification System Employing Image Retrieval Technique", In. Pattern Recognition Letters, Vol. 28, 2367 – 2374, 2007.
- [11] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection", In Proc. Of the CVPR '99, vol. 1, 274-280, 1999.
- [12] M. H. Yang and N. Ahuja, "Detecting human faces in color images ", In International Conference on Image Processing (ICIP), vol 1, 127-130, 1998.
- [13] V. Vezhnevets, V. Sazonov, A. Andreeva, "A Survey on Pixel-based Skin Color Detection Techniques", In Proceedings of the GrapiCon, 85-92, 2003.
- [14] P. Kakumanu, S. Makrogiannis, N. Bourbakis, "A survey of skin-color modeling and detection methods", In *Pattern Recognition*, Vol. 40, pp. 1106-1122, 2007.
- [15] W. Kelly, A. Donnellan and D. Molloy, "Screening for Objectionable Images: A Review of Skin Detection Techniques", In Machine Vision and Image Processing Conference (IMVIP '08), 151 – 158, 2008.
- [16] FTK 3.1 User Guide.

Authors

Jorge Alberto Marcial-Basilio received the BS degree on Electronic and Communications Engineer in 2003 from the National Polytechnic Institute of Mexico. Actually he is a Master degree student at the Mechanical and Electrical Engineering School of the National Polytechnic Institute of Mexico.



Gualberto Aguilar-Torres received the BS degree on Electronic and Communications Engineer; and The MS degree on Microelectronic Engineering, in 2003 and 2005, respectively, from the National Polytechnic Institute of, Mexico. In 2005 he received the Best Thesis award from the National Polytechnic Institute of Mexico for his Master research work and a member of the National Researchers System of Mexico.



Gabriel Sanchez-Perez received the BS degree on Computer Science Engineer; and the PhD degree on Electronic and Communications, in 1999 and 2005, respectively, from the National Polytechnic Institute, Mexico City. From January 2001 to October 2006 he joined the Computer Engineering Department Electrical and Mechanical Engineering School at the National Polytechnic Institute as Assistant Professor. In October 2006 he joins the Graduate School of the National Polytechnic Institute of Mexico where he is now a Professor. Prof. Sanchez-Perez is a member of the IEEE.



Linda Karina Toscano-Medina received the BS degree on Computer Science Engineer; and the PhD degree on Electronic and Communications, in 1999 and 2005, respectively, from the National Polytechnic Institute, Mexico City. From January 2003 to October 2007 she joined the Computer Engineering Department Electrical and Mechanical Engineering School at the National Polytechnic Institute as Assistant Professor. In October 2007 she joins the Graduate School of the National Polytechnic Institute of Mexico where she is now a Professor. Prof. Toscano Medina is a member of the National Researchers System of Mexico



Hector Perez-Meana received the BS Degree in Electronics Engineers from the Universidad Autonoma Metropolitana (UAM) Mexico City in 1981, the M.S. degree from the University of Electro-Communications, Tokyo Japan in March 1986, and a Ph. D. degree in Electrical Engineering from Tokyo Institute of Technology, Tokyo, Japan, in 1989. In 1981 he joined the Electrical Engineering Department of the Metropolitan University where he was a Professor. From March 1989 to September 1991, he was a visiting researcher at Fujitsu Laboratories Ltd, Kawasaki, Japan. In February 1997, he joined the Graduate Department of The Mechanical and Electrical Engineering School on the National Polytechnic Institute of Mexico, where he is now a Professor. In 1991 Prof. Perez-Meana received the IEICE excellent Paper Award, and in 1999 and 2000 the IPN Research Award. In 1998 Prof. Perez-Meana was Co-Chair of the ISITA'98. His principal research interests are signal and image processing, pattern recognition, watermarking, steganography and related fields. Dr. Perez-Meana is a senior member of the IEEE, a member of the IEICE, the IET, the National Researchers System of Mexico and the Mexican Academy of Science.



Enrique Escamilla-Hernandez was born in Hidalgo, Mexico, on November 15th, 1973. He received the B.Sc. degree in Electronic from Metropolitan Autonomous University, DF, Mexico, in 1998 and the M.Sc. degree in microelectronic sciences from IPN, D.F., Mexico, in 2003 and Ph.D. degree in communication and electronic from IPN, D.F., Mexico, in 2006. His research interests are in designing algorithms for robust and intelligible image and speech signal processing for robot applications.



Explicit Image Detection using YCbCr Space Color Model as Skin Detection

JORGE ALBERTO MARCIAL BASILIO¹, GUALBERTO AGUILAR TORRES², GABRIEL SÁNCHEZ PÉREZ³, L. KARINA TOSCANO MEDINA⁴, HÉCTOR M. PÉREZ MEANA⁵

Sección de Estudios de Posgrados e Investigación

¹Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacan
Santa Ana #1000 Col. San Francisco Culhuacan, Del. Coyoacán C.P. 04430

MEXICO CITY

jmarcialb0300@ipn.mx¹, gaguil@ipn.mx², gsanchez@ipn.mx³, ltoscano@ipn.mx⁴,
hmperezm@ipn.mx⁵

Abstract: - In this paper a novel way to detect explicit content images is proposed using the *YCbCr* space color, moreover the color pixels percentage that are within the image is calculated which are susceptible to be a tone skin. The main goal to use this method is to apply to forensic analysis or pornographic images detection on storage devices such as hard disk, USB memories, etc. The results obtained using the proposed method are compared with Paraben's Porn Detection Stick software which is one of the most commercial devices used for detecting pornographic images. The proposed algorithm achieved identify up to 88.8% of the explicit content images, and 5% of false positives, the Paraben's Porn Detection Stick software achieved 89.7% of effectiveness for the same set of images but with a 6.8% of false positives. In both cases were used a set of 1000 images, 550 natural images, and 450 with highly explicit content. Finally the proposed algorithm effectiveness shows that the methodology applied to explicit image detection was successfully proved vs Paraben's Porn Detection Stick software.

Key-Words: - Skin Detection, YCbCr Color Space, Pattern Recognition, Explicit Content

1 Introduction

The information on the Internet is becoming more and more plentiful. The juveniles should be prevented from getting access to adult information, such as adult images, so the development of adult image recognizing technology is urgently desired. However, it is difficult to recognize adult image accurately. So far, adult images can be divided into three categories, which are images of nude body, close-up images of erotogenic parts, and images having pornographic action [1].

Filtering images with adult classified content is very important for searching principal Internet browser programs to avoid offensive content. Nowadays there are some ways to stop pornographic images on computers, such as blocking unwanted sites or identifying images that show explicit content. There are some programs in the foreign market that allow blocking sites on Internet with offensive or explicit content such as: CyberPatrol, ContentProtect, NetNanny, Family.net and K9 Web Protection [2]. All these programs provide parental control to safeguard their children using the Internet. There are some others programs which detect pornographic images within the computer such as: SurfRecon that

offers a program for this purpose and other tool is that the company Paraben offers which its name is Porn Stick Detection [3].

There are some papers on this subject such as: the paper carried out by Forsyth and Fleck who develop software to detect naked people [4], Wiederhold and Wang proposed an algorithm for doubtful content images recognition [5], and Li Chen *et al* design a skin detector based-on Neural Network [6].

In this paper a novel algorithm to detect explicit images is proposed. It is based on image processing, skin detector, and pattern recognition techniques. First the image is transformed to *YCbCr* color space to discriminate all objects into the image that are not of interest. Next the threshold used for skin detection is calculated using this threshold the image is filtered to segment a person or people shape within the image. Finally the image likelihood is estimated to know whether an image with explicit content is or not.

The paper is organized as follows. An introduction of *RGB* and *YCbCr* color spaces and the skin detector method used in this system are shown in sections 2 and 3 respectively. Section 4 shows the proposed system and in the section 5 the results are shown as a comparative with Paraben's Porn Detection Stick. Finally the conclusions are given.

2 Color Spaces

2.1 The RGB Color Space

The *RGB* color space is an additive color model in which the primary colors red, green, and blue light are added together in various ways to reproduce a broad array of colors. The name comes from the initials of the three colors Red, Green, and Blue. The *RGB* color model is shown in the Figure 1.

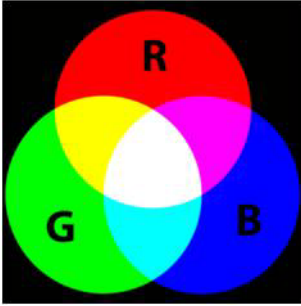


Fig. 1 *RGB* Color Model

The main purpose of the *RGB* color model is for sensing, representation, and display of images in electronic systems, such as televisions and computers.

The *RGB* color model is an additive in the sense that three light beams are added together to make a final color. To form a color with *RGB*, three colored light beams (one red, one green, and one blue) should be superimposed. Each of the three beams is called a component of that color, and each can have arbitrary intensity, from fully off to fully on, in the mixture. Zero intensity for each component gives the darkest color (no light, considered the black), and full intensity of each gives a white.

A color in the *RGB* color model is described by indicating how much of each of the red, green, and blue is included in each component which can vary from zero to a defined maximum value which depends of the application. In computing, the component values are often stored as integer numbers in the range 0 to 255.

2.1 The *YCbCr* Color Space

The *YCbCr* color space is widely used in digital video, image processing, etc. In this format, luminance information is represented by a single component, *Y*, and color information is stored as two color-difference components, *Cb* and *Cr*. Component *Cb* is the difference between the blue component and a reference value, and component *Cr* is the difference between the red component and a reference value.

The *YCbCr* color model was developed as part of ITU-R BT.601 during the development of a worldwide digital component video standard. *YCbCr* is a scaled and offset version of the *YUV* color model. *Y* is the luma component defined to have a nominal 8-bit range of 16 – 235; *Cb* and *Cr* are the blue-difference and red-difference chroma components respectively, which are defined to have a nominal range of 16 – 240.

The transformation used to convert from *RGB* to *YCbCr* color space is shown in the equation (1):

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

In contrast to *RGB*, the *YCbCr* color space is luma-independent, resulting in a better performance. The corresponding skin cluster is given as [7]:

$$Y > 80$$

$$85 < Cb < 135$$

$$135 < Cr < 180,$$

$$\text{Where } Y, Cb, Cr = [0, 255].$$

Chai and Ngan [8] have developed an algorithm that exploits the spatial characteristics of human skin color. A skin color map is derived and used on the chrominance components of the input image to detect pixels that appear to be skin. Working in this color space Chai and Ngan have found that the range of *Cb* and *Cr* most representatives for the skin-color reference map are:

$$77 \leq Cb \leq 127 \quad \text{and} \quad 133 \leq Cr \leq 173$$









However due to that our purpose is to find human skin from different races, the thresholds given above works only with a Caucasian people skin because the first threshold only finds people with white skin, and the second threshold segments people of different places of the world but some pixels are detected as skin but really not. For this reason is proposed a new skin threshold to segment people within the image

regardless skin color, so after exhaustive image histogram analysis, the optimal range threshold was:

$$80 \leq Cb \leq 120 \quad \text{and} \quad 133 \leq Cr \leq 173$$

Some examples of segmentation of people shape using the three thresholds are shown in the Table 1. Kukharev and Novosielski proposed the first threshold, Chai and Ngan proposed the second one, and the third one is proposed in this paper.

Table 1 Comparison between the three thresholds

Original Image	1 st Threshold	2 nd Threshold	3 rd Threshold
			
			

3 Skin Detection

Skin detection can help detect a human limb, torso, or face within a picture. Lately many methods of skin identification within a digital image have been developed. Skin color has proved to be a useful and robust method for face detection, localization and tracking. There have been a number of researchers who have looked at using color information to detect skin. Jones and Rehg [9] constructed a color model using histogram-learning techniques at *RGB* color space. Yang and Auhuja [10] estimated probability density function of human skin color using a finite Gaussian mixture model whose parameters are estimated through the EM algorithm. There are other researchers who have developed papers about the different models of skin detection as Vezhnevets *et al.* [11], Kakumanu *et al.* [12], Kelly *et al.* [13].

In this paper a novel solution using the modified *YCbCr* color space threshold, which is very similar to the *RGB* color space is proposed.

Once the color transformation has been made, the next stage is to proceed to pixel detection with human skin. This was achieved by observing several images, which are a threshold where most people with different skin color within the image can be segmented using the histograms techniques.

To determine the optimal threshold it was necessary to do a histogram analysis in the *YCbCr* color space of many images. The results expressed in Figure 2 and 3 shows the histograms values which help to decide the optimal threshold for different people who have different skin color.

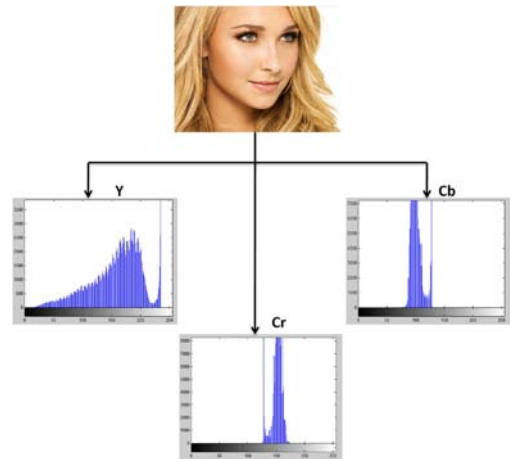


Fig. 2 White Skin Girl

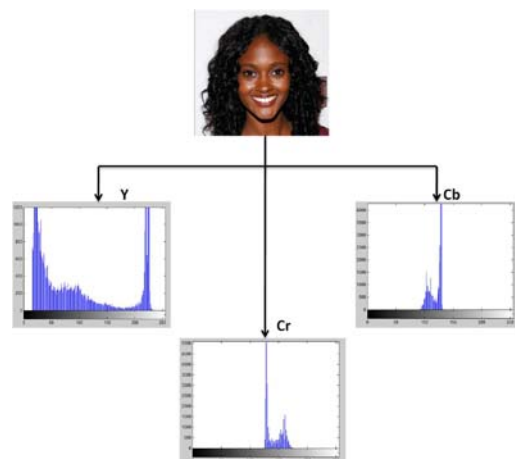


Fig. 3 Black Skin Girl

As observed in the images above, the luminance *Y* histogram is so different for both cases white and black skin because the energy is concentrated in opposite of the histogram, so only the *Cb* and *Cr* chrominance histograms are used to propose a new threshold which includes people with different skin color from white to black skin.

4 Proposed System

Figure 4 shows the proposed system, which has five stages: the image transformation from the *RGB* to *YCbCr* color space is done in the first stage. Next the threshold calculation is done using the chrominance *Cb* and *Cr* histograms analysis. Filter stage or classifier is obtained using the skin detection with the threshold that was obtained in the previous stage.

After the Skin Pixel Quantifier count the pixels number with human skin. Finally the decision stage classify the images by the percentage of skin that the image content.

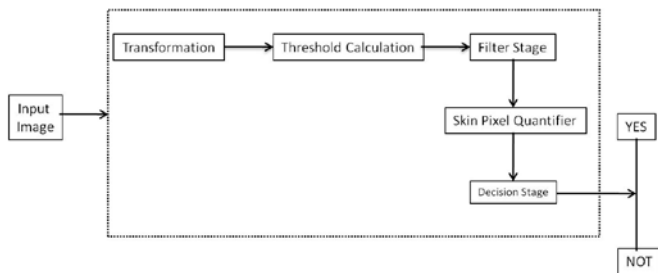


Fig. 4 Proposed System

Skin detection can be used as the basis for detection of the images with explicit content because there is a considerable relationship between the images with large areas of skin and pornographic images or with explicit content.

5 Results

A test to see the performance of the proposed system was using different images from Internet. The system can process different kind of images, as images in different lighting conditions and images with different size.

The input images for testing are classified in: a) images of naked people or with explicit content and b) natural images. In the images of naked people Asians, Caucasians, Europeans, Latin Americans and a little amount of people with black skin can be found. A natural image is one that by its nature does not contain explicit or pornographic content. In the set of natural images there are different kinds of images such as: dressed people, animals, plants, cars, cartoons, landscapes and others were also obtained from Internet.













First the input image is converted from the RGB color space to the YCbCr color space. This discriminates the objects in which we have no interested from the input image. After this stage, skin detection is used to identify the areas that contain some skin tone, and in this way only get the image of the person or people within the input image.

At this point, all skin areas detected are taken, and proceed to count the amount of pixels that there exists within the image to estimate the likelihood that this image is classified as an image with pornographic content or not. The results obtained are shown in Table 2.

To be able determine if the input image has pornographic content, a mathematic expression is proposed, described in (2), and in this way obtain the percentage of amount of skin that there are in the input image. If the percentage is more than 50% it is considered as to have objectionable content.

$$skin\ percentage = \frac{\# skin\ color\ pixels}{\# image\ pixels\ in\ total} \times 100 \quad (2)$$

Table 2 Classification results using the proposed system

Input Image	Images Classification		
	YCbCr Image	Proposed Classifier	Skin Percentage
			60.458 %
			54.049%
			19.765%
			3.323%

For this experiment the set was used had 450 adult images which the majority of images are naked people and a little amount of close-up images of erotogenic parts; and 550 natural images.

Using the system proposed can be proved that it could recognize the 88.8% of all images with 5% of false positives.

This system is used as reference to know if the analyzed images have a certain quantity of pixels with skin color. This is a factor to determine if the image has naked people, as the majority of images that have undressed people are made up of skin zones that take up most of the image.

Within the results, it can be observed that the selected threshold for the image segmentation works efficiently in people with white skin, although to brown or black people, this threshold does not segment the total of skin areas. However, we can estimate its naked people likelihood with the fact that details are not lost detecting such things.

As mentioned in the section 1, also software was used from Paraben's called Porn Stick Detection, to make a comparison between the proposed system and its system to detect images. It works with a software that is within the USB device, only the user select the threshold to start the search within the computer, the threshold is the following: 0 Fewer False Positives and 100 More False Positive; this software search all the images no matter that have been deleted. The images are classified of two different ways: a) Images Highly Suspect, and b) Images Suspect. For these experiment three tests was done using different threshold for every experiment. The results obtained are shown in the Table 3.

Table 3 Results obtained with Paraben's Porn Stick Detection

	1 st Experiment	2 nd Experiment	3 rd Experiment
Threshold	50	65	75
Recognize	64.30%	75.40%	89.10%
False Positive	2.6%	4.9%	6.2%

6 Conclusions

This paper proposed an algorithm to detect images with explicit or pornographic content in color images, using the YCbCr color space and a method of skin detection which works effectively although in some images it could find some errors, due to the image lighting conditions when taken, another factor that can be by a bad interpretation of the system.

The YCbCr color space is an important method to be able to decrease all the lighting problems that the image could be had; this is achieved using the components of chrominance Cb and Cr only.

The proposed system gives an output image that only shows color skin pixels within the image, in basis to this can be known the likelihood that the image is an explicit content or not, due to explicit or pornographic content image in most part has skin color pixels.

The importance of comparison between the proposed system and the tool Paraben's Porn Stick Detection was done to know whether the proposed

system could do the same work, and this way know whether the input image is a pornographic image or not, at final could prove that the proposed system carry out effectively, although the tool has a wide threshold that the proposed system. The proposed system achieved a 88.8% for explicit images detection, compared with 89.1% of Paraben's Porn Detection Stick which is one of the commercial software used for this purpose. It was concluded that the proposed system is reliable and can be used for this purpose.

References:

- [1] Xiaoyin Wang; Changzhen Hu; Shuping Yao; "An Adult Image Recognizing Algorithm Based on Naked Body Detection", Computing, Communication, Control, and Management 2009. (CCCM 2009). ISECS International Colloquium on, pp. 197 – 200, 2009.
- [2] C. Habis, and F. Krsmanovic, "Explicit Image Filter," Stanford Univ., 2005.
- [3] (2010) The Paraben's Porn Stick Detection website. [Online]. Available: <http://parabensticks.com/porn-detection-stick.html>.
- [4] D. A. Forsyth. M. Fleek, and C. Bregler, "Finding naked people", Proc.Forth European Conference on Computer Vision. pp 593-602, 1996
- [5] G. Wiederhold and J. Z. Wang, "WIPE (TM): Wavelet Image Pornography Elimination; A System for Screening Objectionable Images".
- [6] Li Chen, J. Zhou, Z. Liu, W. Chen and G. Xiong, "A Skin Detector Based on Neural Network", In IEEE 2002 International Conference on Communication, Circuits and Systems and West Sino Expositions, vol. 1, 615 – 619, 2002.
- [7] G. Kukharev, A. Novosielski, "Visitor identification elaborating real time face recognition system", In Proc. 12th Winter School on Computer Graphics (WSCG), Plzen, Czech Republic, pp. 157 – 164, Feb. 2004.
- [8] D. Chai, and K.N. Ngan, "Face segmentation using skin-color map in videophone applications". IEEE Trans. on Circuits and Systems for Video Technology, 9(4): 551-564, June 1999.
- [9] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection", In Proc. Of the CVPR '99, vol. 1, 274-280, 1999.
- [10] M. H. Yang and N. Ahuja, "Detecting human faces in color images ", In International Conference on Image Processing (ICIP), vol 1, 127-130, 1998.

- [11] V. Vezhnevets, V. Sazonov, A. Andreeva, “A Survey on Pixel-based Skin Color Detection Techniques”, In Proceedings of the GrapiCon, 85-92, 2003.
- [12] P. Kakumanu, S. Makrogiannis, N. Bourbakis, “A survey of skin-color modeling and detection methods”, In *Pattern Recognition*, Vol. 40, pp. 1106-1122, 2007.
- [13] W. Kelly, A. Donnellan and D. Molloy, “Screening for Objectionable Images: A Review of Skin Detection Techniques”, In Machine Vision and Image Processing Conference (IMVIP ‘08), 151 – 158, 2008.

Detection of Pornographic Digital Images

Jorge A. Marcial-Basilio, Gualberto Aguilar-Torres, Gabriel Sánchez-Pérez, L. Karina Toscano-Medina, and Héctor M. Pérez-Meana

Abstract— In this paper a novel algorithm to detect explicit content or pornographic images is proposed using the transformation from the RGB model color to the YCbCr or HSV color model, moreover using the skin detection the image is segmented, finally the percentage of pixels that was detected as skin tone is calculated. The results obtained using the proposed algorithm are compared with two software solutions, Paraben's Porn Detection Stick and FTK Explicit Image Detection, which are the most commercial software solutions to detect pornographic images. A set of 800 images, which 400 pornographic images and 400 natural images, is used to test each system. The proposed algorithm carried out identify up to 68.87% of the pornographic images, and 14.25% of false positives, the Paraben's Porn Detection Stick achieved 71.5% of recognizing but with 33.5% of false positives, and FTK Explicit Image Detection achieved 69.25% of effectiveness for the same set of images but 35.5% of false positives. Finally the proposed algorithm works effectively to carry out the main goal which is to apply this method to forensic analysis or pornographic images detection on storage devices.

Keywords—Explicit Content, Pattern Recognition, Skin Detection, The YCbCr and the HSV color models.

I. INTRODUCTION

THE information on the Internet is becoming more and more plentiful. The juveniles should be prevented from getting access to adult information, such as adult images, so the development of adult image recognizing technology is urgently desired. However, it is difficult to recognize adult image accurately. So far, adult images can be divided into three categories, which are images of nude boy, close-up images of erotogenic parts, and images having pornographic action [1]. This kind of media is also available for children and is an increasingly problem for many parents.

Filtering images with adult classified content is very

Jorge A. Marcial-Basilio is a Master degree student of the National Polytechnic Institute of Mexico, e-mail: jmarcialb0300@ipn.mx

Gualberto Aguilar-Torres is a Professor at Graduate Department of the National Polytechnic Institute of Mexico, e-mail: gaguilar@ipn.mx

Gabriel Sánchez-Pérez is a Professor at Graduate Department of the National Polytechnic Institute of Mexico, e-mail: gsanchez@ipn.mx

L. Karina Toscano-Medina is a Professor at Graduate Department of the National Polytechnic Institute of Mexico, e-mail: ltoscano@ipn.mx

Héctor M. Pérez-Meana is a Professor at Graduate Department of the National Polytechnic Institute of Mexico, e-mail: hmperez@ipn.mx

important for searching principal Internet browsers programs to avoid offensive content [2]. Nowadays there are some ways to stop pornographic images on computers, such as blocking unwanted sites or identifying images that show explicit content. There are some programs in the foreign market that allow blocking sites on Internet with offensive or explicit content such as: CyberPatrol, ContentProtect, NetNanny, Family.net and K9 Web Protection [3]. All these programs provide parental control to safeguard their children using the Internet. There exists some others programs which detect pornographic images within the computer such as: SurfRecon that offers a program for this purpose, Paraben's Porn Detection Stick [4] is another software, the company Access Data offers a software solution for informatics forensic call Forensic Toolkit (FTK 3.1), this software have a tool FTK Explicit Image Detection for this use.

There are some papers on this subject such as: the carried out by Forsyth and Fleck who developed software to detect naked people [5], Wiederhold and Wang proposed an algorithm for doubtful content images recognition [6], and Li Chen *et al* design a skin detector based-on Neural Network [7], P. Fuangkhan and T. Tanprasert develop a system to detect obscene pictures using nipple detection [8].

In this paper a novel algorithm to detect explicit images is proposed using two different color models YCbCr and HSV and compare with two software solutions Paraben's Porn stick Detection and FTK Explicit Image Detection. It is based on image processing, skin detector, and pattern recognition techniques. First the image is transformed to YCbCr or HSV color model to discriminate all objects into the image that are not of interest. Next the threshold used for skin detection is calculated for each color model, using this threshold the image is filtered to segment a person or people shape within the image. Finally the image likelihood is estimated to know whether an image with explicit content is or not.

The paper is organized as follows. The Proposed System and an introduction of the RGB, HSV and YCbCr color models are shown in sections 2 and 3 respectively. Section 4 shows the threshold calculation, in the section 5 Skin Detection is shown. Image Classification is shown in section 6. The results are shown as a comparative between the proposed system using the two color models, and the systems Paraben's Porn Stick Detection and FTK Explicit Image Detection in Section 7. Finally the conclusions are given.

II. PROPOSED SYSTEM

The Figure 1 shows the proposed system, which has four stages: the image transformation from RGB to YCbCr, or HSV color model is done in the first stage. Filter stage is obtained using the skin detection with the proposed threshold. Next a Quantifier is used to count the number of skin pixels. Finally in the decision stage the images are classified by the percentage of skin that contains the image.

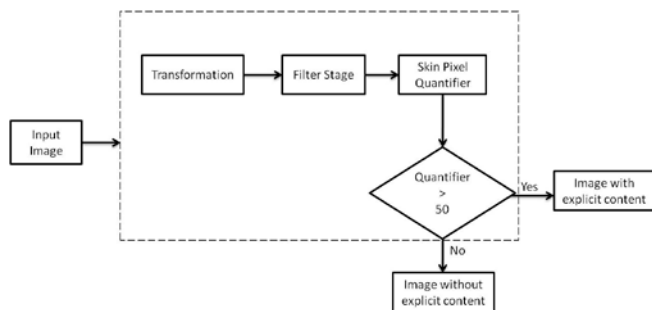


Fig. 1 Proposed System

To determine the threshold of our proposed system, a histogram analysis was performed. Figure 2 shows this stage.



Fig. 2 Algorithm to Threshold Calculation

III. COLOR MODELS

A. The RGB Color Model

The RGB color model is an additive color model in which the primary colors red, green, and blue light are added together in various ways to reproduce a broad array of colors. The name comes from the initials of the three colors Red, Green, and Blue. The RGB color model is shown in the Figure 3.

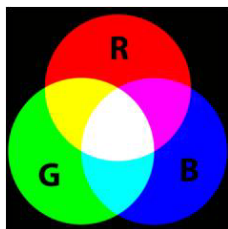


Fig. 3 RGB Color Model

The main purpose of the RGB color model is for sensing, representation, and display of images in electronic systems, such as televisions and computers.

The RGB color model is an additive in the sense that three light beams are added together to make a final color. To form a color with RGB, three colored light beams (one red, one green, and one blue) should be superimposed. Each of the three beams is called a component of that color, and each can

have arbitrary intensity, from fully off to fully on, in the mixture. Zero intensity for each component gives the darkest color (no light, considered the black), and full intensity of each gives a white.

A color in the RGB color model is described by indicating how much of each of the red, green, and blue is included in each component which can vary from zero to a defined maximum value which depends of the application. In computing, the component values are often stored as integer numbers in the range 0 to 255.

B. YCbCr Color Model

The YCbCr color model is widely used in digital video, image processing, etc. In this format, luminance information is represented by a single component, Y, color information is stored as two color-difference components, Cb and Cr. Component Cb is the difference between the blue component and a reference value, and component Cr is the difference between the red component and a reference value.

The YCbCr color model was developed as part of ITU-R BT.601 during the development of a world-wide digital component video standard. YCbCr is a scaled and offset version of the YUV color model. Y is the luma component defined to have a nominal 8-bit range of 16 – 235; Cb and Cr are the blue-difference and red-difference chroma components respectively, which are defined to have a nominal range of 16 – 240.

The transformation used to convert from RGB to YCbCr color space is shown in the equation (1):

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.996 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

The Figure 4 shows the transformation of the input image using the expression above.

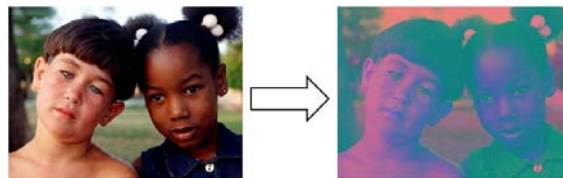


Fig. 4 Transformation from RGB to YCbCr color model

C. HSV Color Model

HSV color model (Hue, Saturation, and Value) is a non-linear transformation of the RGB color model, and the colors are the combination of the three values: the Hue (H), Saturation or color quantity (S), and itself value (V) [9]. These values are represented in a circular diagram, as shown in Figure 5.

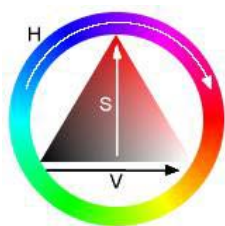


Fig. 5 HSV Color Model

The three magnitudes can have the following values:

Hue: The type of color (e.g. red, green, or yellow). These are represented as a degree of angle whose possible values range from 0° to 360° (although for some applications are normalized from 0 to 100%).

Saturation: Is represented as a distance from the axis of the black-white glow. The possible values range from 0 to 100%.

Value: Represents the height in the black-white axis. The possible values range from 0 to 100%. 0 is always black. Depending on the saturation, 100 could be white or a more or less saturated color.

The transformation of an input image in RGB color model is achieved using the expressions (2) to (4) that are shown below.

$$H = \arccos \frac{1/2[(R-G)+(R-B)]}{\sqrt{[(R-G)^2 + (R-B)(G-B)]}} \quad (2)$$

$$S = 1 - 3 \frac{\min(R, G, B)}{R + G + B} \quad (3)$$

$$V = \frac{1}{3}(R + G + B) \quad (4)$$

The Figure 6 is a sample of the conversion to the HSV color model of an image in RGB color model.

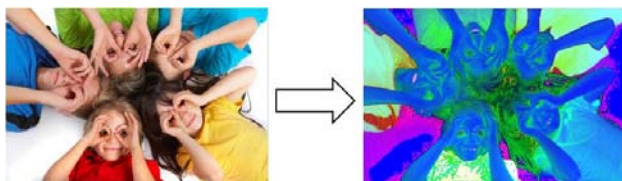


Fig. 6 Transformation from RGB to HSV color model

IV. THRESHOLD CALCULATION

A. YCbCr Histogram Analysis

In contrast to RGB, the YCbCr color model is luma-independent, resulting in a better performance. The corresponding skin cluster is given as [10]:

$$\begin{aligned} Y &> 80 \\ 85 &< Cb < 135 \\ 135 &< Cr < 180, \\ \text{Where } Y, Cb, Cr &= [0, 255]. \end{aligned}$$

N. Sarris *et al.* [11] used only the chrominance components, Cb and Cr, to detect faces in color images, V. Neagoe and M. Neghina [12] proposed a system to face detection using this color model.

Chai and Ngan [13] have developed an algorithm that exploits the spatial characteristics of human skin color. A skin color map is derived and used on the chrominance components of the input image to detect pixels that appear to be skin. Working in this color model Chain and Ngan have found that range of Cb and Cr most representatives for the skin-color reference map are:

$$77 \leq Cb \leq 127 \quad \text{and} \quad 133 \leq Cr \leq 173$$

However due to that our purpose is to find human skin from different races, the thresholds given above works only with a Caucasian people skin because the first threshold only finds people with white skin, and the second threshold segments people of different places of the world but some pixels are detected as skin but really not. For this reason is proposed a new skin threshold to segment people within the image regardless skin color.

A histogram analysis in the YCbCr color model of many images was necessary to determine the optimal threshold. The results expressed in Figure 7 and 8 shows the histograms values which help to decide the best possible threshold for different people who have different skin color.

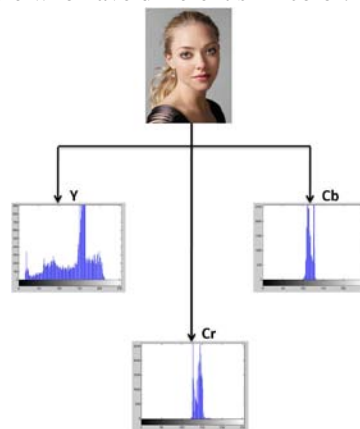


Fig. 7 White Skin Girl

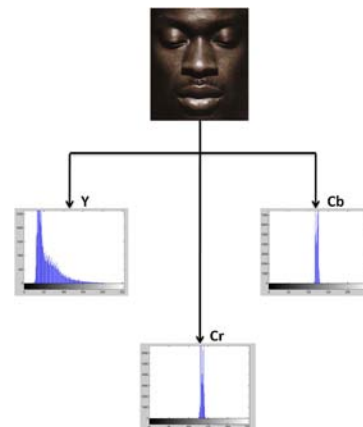


Fig. 8 Black Skin Man

As observed in the images above, the luminance Y histogram is so different for both cases white and black skin because the pixel values are concentrated in opposite of the histogram, so only the Cb and Cr chrominance histograms are used to propose a new threshold which includes people with different skin color from white to black skin.

After exhaustive image histogram analysis, the optimal range threshold was:

$$80 \leq Cb \leq 120 \quad \text{and} \quad 133 \leq Cr \leq 173$$

B. HSV Histogram Analysis

I. Aldasouqui and M. Hassan [14] designed an algorithm to human face detection using this color model.

A histogram analysis in the HSV color model was performed to determine the threshold. Figure 9 and 10 are images with white and black skin. These histograms were necessary to identify the values range for our purpose, i.e. to segment a person or people in one image.

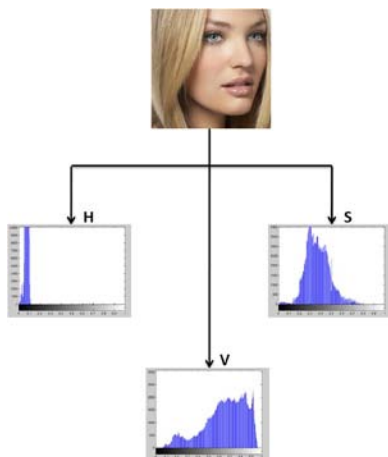


Fig. 9 White Skin Girl

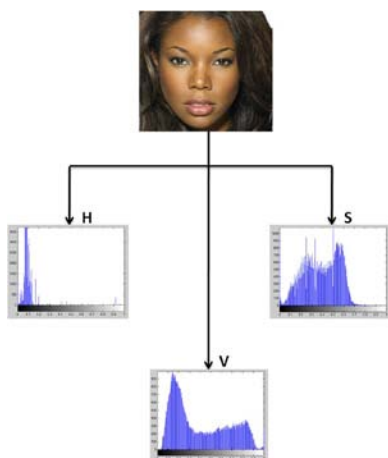


Fig. 10 Black Skin Girl

The histograms observed previously show the values that have the images of people with different races. After an extensive analysis the threshold decided was the following:

$$0 < H < 0.25$$

$$0.15 < S < 0.9$$

$$0.2 < V < 0.95$$

Where H,S, and V are in the range from 0 to 1.







V. SKIN DETECTION

Skin detection can help detect a human limb, torso, or face within a picture. Lately many methods of skin identification within a digital image have been developed. Skin color has proved to be a useful and robust method for face detection, localization and tracking. There have been a number of researchers who have looked at using color information to detect skin. Jones and Rehg [15] constructed a color model using histogram-learning techniques at RGB color model. Yang and Auhuja [16] estimated probability density function of human skin using a finite Gaussian mixture model whose parameters are estimated through the EM algorithm. There are other researchers who have developed papers about the different models of skin detection as Vezhnevets *et al.*[17] Kakumanu *et al.*[18], and Kelly *et al.*[19].

Once the color transformation has been made, the next stage is to proceed to pixel detection with human skin. This was carried out by using the thresholds proposed in this paper, which works in different color models.







The Table I show some example of the skin detection using the proposed thresholds for images of different races of people.

Table I Skin Detection using the proposed threshold to YCbCr color model

Input Image	YCbCr threshold
	
	
	

In the Table I, can be observed that the proposed threshold to the YCbCr color model works efficiently in people with different skin color.

Table II Skin Detection using the proposed threshold to HSV color model

Input Image	HSV threshold
	
	
	

The Table II shows the segmentation in basis to human skin color using the proposed threshold to the HSV model color. This table contains some examples of people with white, brown and black skin, the proposed threshold works correctly.






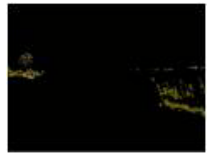






VI. IMAGE CLASSIFICATION

To determine if the image has a explicit content or pornographic, the Skin Pixel Quantifier counts the pixels that the previous stage detect as skin color, then using (4) the skin percentage is calculated to know if the image have objectionable content.

$$skin\ percentage = \frac{\#skin\ color\ pixels}{\#image\ pixels\ in\ total} \times 100 \quad (4)$$

To consider an image as explicit content or pornographic image, the skin percentage must be more than 50% or equal. The Table III shows some examples of the image classification using all the proposed system to classify the images.

Table III Image Classification

Input Image	Images Classification		
	Color Model	Proposed Classifier	Skin Percentage
	YCbCr		60.64%
	YCbCr		56.36%
	YCbCr		2.39%
	HSV		75.55%
	HSV		28.17%
	HSV		39.35%

In the Table III can be seen some examples using the two methods to detect explicit content or pornographic images.

Skin detection can be used as the basis for detection of the images with explicit content because there is a considerable relationship between the images with large areas of skin and pornographic images or with explicit content.

VII. RESULTS

A test to see the performance of the proposed system was using different images from Internet. The system can process different kind of images, as images in different lighting conditions and images with different size.

The input images for testing are classified in: a) images of naked people or with explicit content and b) natural images. In the images of naked people Asians, Caucasians, Europeans, Latin Americans and a little amount of people with black skin can be found. A natural images one that by its nature does not contain explicit or pornographic content. In the set of natural images there are different kinds of images such as: dressed people, animals, plants, cars, cartoons, landscapes and others were also obtain from Internet.

First the input image is converted from RGB model color to YCbCr or HSV color model. After this stage, skin detection is used to identify the areas that contain some skin tone, and in this way only get the image of the person or people within the input image.

At this point, all skin areas detected are taken, and proceed to count the amount of pixels that there exists within the image to estimate the likelihood that this image is classified as an image with pornographic content or not.

For this test, the set was used had 400 adult images which all are of naked people, and 400 natural images.

Using the proposed system was obtained the following results that show in the Table IV:

Table IV Results of Proposed system

Color Model	Pornographic Images Recognizing	False Positives
YCbCr	68.87 %	14.25 %
HSV	68.25 %	18.25 %

The Table IV shows the percentage of explicit content or pornographic images recognizing using the two methods which are using the proposed thresholds to skin detection.

As mentioned in the section 1, there exists software to detect explicit content images which are Paraben’s Porn Detection Stick and FTK 3.1 Explicit Image Detection that are used to make a comparison between the proposed system and these software solutions.

Paraben’s Porn Detection Stick is a thumb drive device that will search through all the images on your computers, scan them for pornographic content, and create a report of suspected pornographic images [4]. It works with software that is within the USB device, only the user selects the sensitivity to start the search within the computer, the range of the sensitivity is the following: 0 Fewer False Positives to 100 More False Positives; The images are classified in Highly Suspect, to compare the proposed system with this software, three test was done using different thresholds for each

experiment. The results obtained are shown in the Table V.

Table V Results obtained with Paraben’s Porn Stick Detection

	1 st Experiment	2 nd Experiment	3 rd Experiment
Sensitivity	65	75	100
Recognize	66.25 %	68.25%	71.50%
False Positives	23.25 %	28.75%	33.50%

Also software was used from Access Data is called Forensic Toolkit 3.1 Explicit Image Detection, to make another comparison between the proposed system and its system to detect images. It makes a forensic image of the device that have to analyze, after the analysis began with three algorithms to determine the percentage of explicit content of an image. The description of the three algorithms used for evaluate the images was obtained from [20] and was shown in the Table VI.






Table VI Explicit Image Detection Profile Types

Profile Name	Level	Description
X-DFT	Default (XS1)	This is the most generally accurate, it is always selected.
X-FST	Fast (XTB)	This is the fastest. It scores a folder by the number of files it contains that meets the criteria for a high likelihood of explicit material. It is built on a different technology than X-DFT and does not use “regular” DNAs. It is designed for very high volumes, or real-time page scoring. Its purpose is to quickly reduce, or filter, the volume of data to a meaningful set.
X-ZFN	Less False Negatives (XT2)	This is a profile similar to S-FST but with more features and with fewer false negatives than X-DFT. Apply this filter after initial processing to all evidence, or to only the folders that score highly using the X-FST option. Checkmark or highlight those folders to isolate them for Additional Analysis.

To test its system, the same set of input images was used using three profiles that have. Some examples of the results

that were obtained using FTK Explicit Image Detection are shown in the Table VII.

Table VII Results obtained with FTK 3.1

Input Image	Algorithms used		
	X-FST	X-DFT	X-ZFN
	93	21	93
	99	11	99
	99	42	99
	11	0	11
	11	11	11

The Table VII shows the explicit content percentages of an image, knowing this the results of the recognizing are shown in the Table VIII. An image is considered as explicit content or pornographic image if two of the three algorithms result with more than 80.

Table VIII Percentage of Recognizing

Recognize	False Positives
69.25 %	35.50%

Other test to prove the efficiency of the system, the system was proved using 100 pornographic images, and the results that was obtained, are shown in the Table IX.

Table IX Results of False Negatives

System	Recognize	False Negatives
Proposed System		
HSV color model	76%	24%
Paraben's Porn Detection Stick	86%	14%
FTK 3.1 Explicit Image Detection	74%	26%

VIII. CONCLUSION

This paper proposed an algorithm to detect explicit content or pornographic in color images, using the YCbCr or HSV color model and a method of skin detection which works effectively although in some images it could find some errors, due to the image lighting conditions when was taken.

The YCbCr and HSV color models are an important method to be able to decrease all the lighting problems that the image could be had; this is achieved using components of chrominance Cb and Cr only for the YCbCr color model, in the case of the HSV color model, all the components are taken to decrease this problem.

The importance of comparing the proposed system with the two software solutions was done to know whether the proposed system could do the same work as the software solutions offer, and this way know whether the input image is a pornographic image or not, at final could prove that the proposed system carry out effectively, although the tools have a wide threshold that the proposed system. Even so the proposed system achieved a 76.25% and 77.75% to the YCbCr and HSV color model respectively, for explicit images detection.

REFERENCES

- [1] Xiaoyin Wang; Changzhen Hu; Shuping Yao; "An Adult Image Recognizing Algorithm Based on Naked Body Detection", Computing, Communication, Control, and Management 2009. (CCCM 2009). ISECS International Colloquium on, pp. 197 – 200, 2009.
- [2] Jorge A. Marcial Basilio, Gualberto Aguilar Torres, Gabriel Sánchez Pérez, L. Karina Toscano Medina, Héctor M. Pérez Meana, "Explicit Image Detection using YCbCr Space Color Model as Skin Detection", In Proc. Of 5th WSEAS Int. Conf. on Computer Engineering and Applications (CEA '11), pp. 123 – 128, 2011.
- [3] C. Habis, and F. Krsmanovic, "Explicit Image Filter," Stanford Univ., 2005.
- [4] (2011) The Paraben's Porn Stick Detection website. [Online]. Available: <http://paraben-sticks.com/porn-detection-stick.html>.

- [5] D. A. Forsyth, M. Fleek, and C. Bregler, "Finding naked people", Proc. Forth European Conference on Computer Vision. pp 593-602, 1996.
- [6] G. Wiederhold and J. Z. Wang, "WIPE (TM): Wavelet Image Pornography Elimination; A System for Screening Objectionable Images".
- [7] Li Chen, J. Zhou, Z. Liu, W. Chen and G. Xiong, "A Skin Detector Based on Neural Network", In IEEE 2002 International Conference on Communication, Circuits and Systems and West Sino Expositions, vol. 1, 615 – 619, 2002.
- [8] P. Fuangkhan and T. Tanprasert, "Nipple Detection for Obscene Pictures", In Proc. Of WSEAS Int. Conf. on Signal, Speech and Image Processing, pp. 315 – 320, 2005.
- [9] Jorge A. Marcial Basilio, Gualberto Aguilar Torres, Gabriel Sánchez Pérez, L. Karina Toscano Medina, Héctor M. Pérez Meana, and Enrique Escamilla Hernández, "Explicit Content Image Detection", In Proc. Of Signal & Image Processing: An International Journal (SIPIJ) Vol. 1, No. 2, pp. 61-72, 2010.
- [10] G. Kukharev, A. Novosielski, "Visitor identification elaborating real time face recognition system", In Proc. 12th Winter School on Computer Graphics (WSCG), Plzen, Czech Republic, pp. 157 – 164, Feb. 2004.
- [11] N. Sarris, N. Grammalidis, and M. Strintzis, "Detection of Faces and Facial Features in Images using a Novel Neural Network Technique", In Proc. Of WSEAS Int. Conf. on Neural Network and Applications, pp. 6361-6366, 2001.
- [12] Victor-Emil Neagoie, Mihai Neghina, "Face Detection Using a Dual Cross-Validation of Chrominance/Luminance Channel Decisions and Decorrelation of the Color Space", In Proc. Of 14th WSEAS Int. Conf. on Computers, pp. 391-396, 2010.
- [13] D. Chai, and K.N. Ngan, "Face segmentation using skin-color map in videophone applications". IEEE Trans. on Circuits and Systems for Video Technology, 9(4): 551-564, June 1999.
- [14] Iyad Aldasouqi, and Mahmoud Hassan, "Human Face Detection System Using HSV", In Proc. Of 9th WSEAS Int. Conf. on Circuits, Systems, Electronics, Control & Signal Processing (CSECS'10), pp. 13-16, 2010.
- [15] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection", In Proc. Of the CVPR '99, vol. 1, 274-280, 1999.
- [16] M. H. Yang and N. Ahuja, "Detecting human faces in color images", In International Conference on Image Processing (ICIP), vol 1, 127-130, 1998.
- [17] V. Vezhnevets, V. Sazonov, A. Andreeva, "A Survey on Pixel-based Skin Color Detection Techniques", In Proceedings of the GrapiCon, 85-92, 2003.
- [18] P. Kakumanu, S. Makrogiannis, N. Bourbakis, "A survey of skin-color modeling and detection methods", In *Pattern Recognition*, Vol. 40, pp. 1106-1122, 2007.
- [19] W. Kelly, A. Donnellan and D. Molloy, "Screening for Objectionable Images: A Review of Skin Detection Techniques", In Machine Vision and Image Processing Conference (IMVIP '08), 151 – 158, 2008.
- [20] FTK 3.1 User Guide.

Jorge Alberto Marcial-Basilio received the BS degree on Electronic and Communications Engineer in 2009 from the National Polytechnic Institute of Mexico. Actually he is a Master degree student at the Mechanical and Electrical Engineering School of the National Polytechnic Institute of Mexico.

Gualberto Aguilar-Torres received the BS degree on Electronic and Communications Engineer; and The MS degree on Microelectronic Engineering, in 2003 and 2005, respectively, from the National Polytechnic Institute of, Mexico. In 2005 he received the Best Thesis award from the National Polytechnic Institute of Mexico for his Master research work and a member of the National Researchers System of Mexico.

Gabriel Sanchez-Perez received the BS degree on Computer Science Engineer; and the PhD degree on Electronic and Communications, in 1999 and 2005, respectively, from the National Polytechnic Institute, Mexico City. From January 2001 to October 2006 he joined the Computer Engineering Department Electrical and Mechanical Engineering School at the National Polytechnic Institute as Assistant Professor. In October 2006 he joins the Graduate School of the National Polytechnic Institute of Mexico where he is now a Professor. Prof. Sanchez-Perez is a member of the IEEE.

Linda Karina Toscano-Medina received the BS degree on Computer Science Engineer; and the PhD degree on Electronic and Communications, in 1999 and 2005, respectively, from the National Polytechnic Institute, Mexico City. From January 2003 to October 2007 she joined the Computer Engineering Department Electrical and Mechanical Engineering School at the National Polytechnic Institute as Assistant Professor. In October 2007 she joins the Graduate School of the National Polytechnic Institute of Mexico where she is now a Professor. Prof. Toscano Medina is a member of the National Researchers System of Mexico.

Hector M. Perez-Meana received the BS Degree in Electronics Engineers from the Universidad Autonoma Metropolitana (UAM) Mexico City in 1981, the M.S. degree from the University of Electro-Communications, Tokyo Japan in March 1986, and a Ph. D. degree in Electrical Engineering from Tokyo Institute of Technology, Tokyo, Japan, in 1989. In 1981 he joined the Electrical Engineering Department of the Metropolitan University where he was a Professor. From March 1989 to September 1991, he was a visiting researcher at Fujitsu Laboratories Ltd, Kawasaki, Japan. In February 1997, he joined the Graduate Department of The Mechanical and Electrical Engineering School on the National Polytechnic Institute of Mexico, where he is now a Professor. In 1991 Prof. Perez-Meana received the IEICE excellent Paper Award, and in 1999 and 2000 the IPN Research Award. In 1998 Prof. Perez-Meana was Co-Chair of the ISITA'98. His principal research interests are signal and image processing, pattern recognition, watermarking, steganography and related fields. Dr. Perez-Meana is a senior member of the IEEE, a member of the IEICE, the IET, the National Researchers System of Mexico and the Mexican Academy of Science.