

RECONOCIMIENTO FACIAL UTILIZANDO VIOLA-JONES Y PATRONES BINARIOS

Hirvin Gonzalez , Sergio Velásquez.

{hirvin17, sergiovelasquezg}@gmail.com

Universidad Nacional Experimental Politécnica “Antonio José de Sucre” (UNEXPO) Vicerectorado Puerto Ordaz
Centro de Investigación de Redes Neuronales Artificiales y Robotica (CIRNAR)

Recibido (09/05/19), Aceptado (13/05/19)

Resumen: En este artículo se muestra las capacidades que posee un sistema de reconocimiento de reconocimiento facial, dado diferentes condiciones que se pueden presentar en cualquier espacio físico. El diseño cuenta con sistema que es capaz de reconocer el rostro de una persona y mostrar el porcentaje de acierto que tiene con respecto a las imágenes del rostro de usuario que sirven para que el sistema pueda identificar a la persona. Utilizando la librería OpenCV de Python mediante el algoritmo de Viola-Jones y el algoritmo de Patrones Binarios Locales (LBP), se realizó un interfaz que es capaz de identificar a tantos usuarios se encuentren registrados en la base de datos.

Palabras Claves: Reconocimiento Facial, Vision Artificial, OpenCV, Machine Learning, Procesamiento de imágenes, Python.

FACIAL RECOGNITION USING VIOLA-JONES AND BINARY PATTERNS

Abstract: This article shows the capabilities of a facial recognition recognition system, given different conditions that can occur in any physical space. The design has a system that is able to recognize the face of a person and show the percentage of success that it has with respect to the user face images that serve so that the system can identify the person. Using the OpenCV library of Python using the algorithm of Viola-Jones and the algorithm of Local Binary Patterns (LBP), an interface was made that is able to identify so many users are registered in the database.

Keywords: Facial Recognition, Artificial Vision, OpenCV, Machine Learning, Image Processing, Python

I. INTRODUCCIÓN

La tecnología de reconocimiento facial no es precisamente un fenómeno nuevo. Los primeros experimentos con esta tecnología se remontan a la década de 1960, aunque, en aquel entonces, la investigación para comenzar con su desarrollo se mantuvo en secreto. Si bien hoy en día la ciencia detrás del software es mucho más matemática, automática y cuenta con el respaldo de los potentes equipos sofisticados actuales, los primeros modelos requerían un mayor nivel de intervención humana y, por lo tanto, eran automatizados solo en forma parcial.

Su uso en estos casos es relativamente sencillo: una cámara recopila las imágenes, mediante el uso de un algoritmo informático, el sistema trata de identificar al usuario; para ello mide ciertos rasgos de su rostro, como la distancia entre sus ojos o el ancho de la nariz. El software de reconocimiento facial se encuentra plenamente integrado ya en muchos mercados, desde el de la seguridad hasta el de la domótica. Sin embargo, es poco habitual que ese tipo de software tenga la cate-

ría de código abierto.

Por consiguiente, partiendo de este tipo de sistema se realizará mediante el uso de código abierto y el cual se podrá visualizar en tiempo real la identificación del usuario con su respectivo porcentaje de acierto y también se mostrará la hora actual en la que se está haciendo la identificación.

II. DESARROLLO

El diseño está basado en un sistema de reconocimiento facial, el cual será aplicado a un entorno de trabajo. Este sistema contará con una cámara web para poder capturar los patrones faciales de los usuarios previamente registrados, los cuales serán procesados en una computadora. Dicho programa almacenará los patrones en la base de datos, realizará el entrenamiento para reconocer a los usuarios y tendrá una interfaz que será capaz de visualizar e identificar a los usuarios. También se enviará mediante un correo electrónico quien ingreso en el sistema y a qué fecha y hora fue que ingreso. Una vez el usuario sea identificado, este sistema será capaz

de dar acceso al entorno de trabajo y poder controlar la temperatura, la iluminación y el encendido de los equipos electrónicos según los requerimientos de cada uno de los usuarios.

A.Reconocimiento Facial

El sistema de reconocimiento facial es una aplicación que se encarga de identificar automáticamente a una persona dirigida por un ordenador, el cual realiza un análisis de las características faciales del usuario adquiridas mediante una imagen o una fuente de video comparándolas con una base de datos.

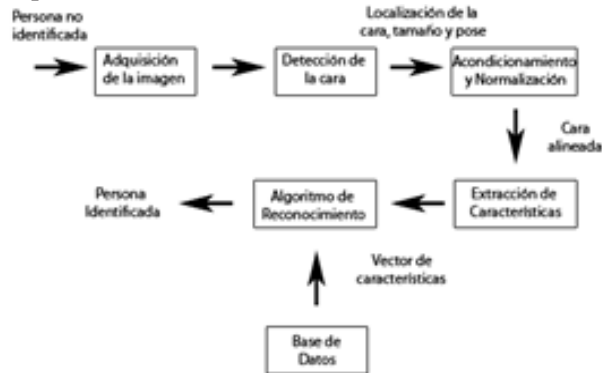


Figura 1. Sistema de reconocimiento facial

Estos sistemas se dividen en cuatro etapas. La primera etapa es la detección del rostro y su respectiva localización en la imagen. La segunda etapa es el procesado para escalar, recortar y ecualizar la imagen. La tercera etapa es la extracción de características la cual proporciona la información para distinguir la cara. La cuarta etapa es el reconocimiento del usuario utilizando los valores obtenidos y compáralo con los de la base de datos.

Estas cuatro etapas se aplican a cada usuario que queramos reconocer. No obstante, existe una etapa previa de entrenamiento para preparar el sistema que solo se ejecuta una vez. En esta etapa, las tres primeras fases se aplican a cada uno de los usuarios para extraer un conjunto de entrenamiento para extraer las características más relevantes de dichas imágenes y almacenarla en la base de datos. Una vez entrenado el sistema, se podrá reconocer a todos los usuarios registrados.

1)Detección Facial

La detección facial es el proceso en el que el software determina, mediante algoritmos, si hay rostros humanos en una foto o vídeo. No determina la identidad de una persona, tan solo determina si hay alguna cara. En la actualidad existen distintos algoritmos para realizar una detección facial, muchos de ellos se utilizan para un área en específica y puede ser más complejo dependiendo de su utilidad. Por lo que para este caso se utilizara el

algoritmo de Viola-Jones el cual permite detectar a los usuarios en tiempo real.

a)Algoritmo de Viola-Jones

El algoritmo de Viola-Jones es un algoritmo de detección de rostros con un coste computacional bajo que permite que sea empleado en tiempo real, propuesto por Paul Viola, de Mitsubishi Electric Research Labs, y Michael Jones, de Compaq CRL, en julio de 2001. Su desarrollo fue motivado por el problema de la detección de caras, donde sigue siendo ampliamente utilizado, pero puede aplicarse a otras clases de objetos que, como las caras, estén caracterizados por patrones típicos de iluminación.

El algoritmo se basa en la comparación entre las intensidades luminosas de regiones rectangulares de las imágenes denominadas Características Haar-Like que calcula empleando una imagen integral. Estos clasificadores, que por sí mismos tienen una probabilidad de acertar solo ligeramente superior a la del azar, se agrupan en una cascada empleando un algoritmo de aprendizaje basado en AdaBoost para conseguir un alto rendimiento en la detección, así como una alta capacidad discriminativa en las primeras etapas.

(1)Características Haar Like

Haar-Like es una característica simple rectangular que se usa como una característica de entrada para el clasificador en cascada. En la figura 2, hay algunos filtros basados en la característica Haar-Like. Aplicando cada uno de estos filtros en uno. En el área especial de la imagen, las sumas de píxeles debajo de las áreas blancas se restan de las sumas de píxeles debajo de las áreas negras. Es decir, el peso del área blanca y negra se puede considerar como "1" y "-1", respectivamente.

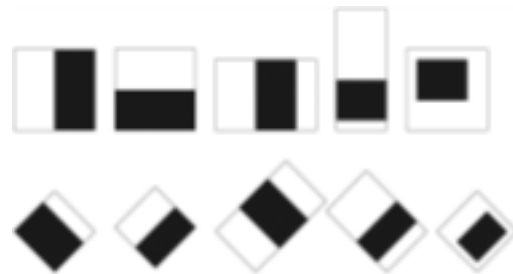


Figura 2. Filtros basados en características Haar-like

Estas características son rasgos muy simples que se buscan en las imágenes y que consisten en la diferencia de intensidades luminosas entre regiones rectangulares adyacentes. Las características por tanto quedan definidas por unos rectángulos y su posición relativa a la ventana de búsqueda y adquieren un valor numérico resultado de la comparación que evalúan. En el trabajo

de Viola-Jones, las características se definen sobre una ventana de búsqueda básica de 24x24 píxeles, lo que da lugar a más de 180000 características posibles.

(2) Imagen Integral

Imagen integral es un método rápido para calcular la característica Haar-Like. La imagen integral es la suma de todos los valores de píxeles en la parte superior e izquierda de la posición (x, y). El área negra de la figura 3 muestra la imagen integral. Con la imagen integral, la característica Haar-Like se puede calcular rápidamente mediante una simple suma y resta.



Figura 3. Imagen integral de un punto

La localización x, y, contiene la suma de los píxeles de la parte superior izquierda de la imagen y se puede calcular como se indica a continuación

$$II(x, y) = \sum_{x' \leq x, y' \leq y} Im(x', y') \quad (1)$$

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

Algoritmo AdaBoost

El algoritmo de AdaBoost (un meta-algoritmo de machine learning) para elegir características y mejorar el rendimiento se usa repetidamente. AdaBoost es un método de clasificación que combina varios clasificadores básicos para formar un único clasificador más complejo y preciso. La idea se basa en la afirmación de que varios clasificadores sencillos, cada uno de ellos con una precisión ligeramente superior a una clasificación aleatoria, pueden combinarse para formar un clasificador de mayor precisión, siempre y cuando se disponga de un número suficiente de muestras de entrenamiento.

El algoritmo Viola Jones utiliza el AdaBoost en la forma en que mezcla una serie de clasificadores AdaBoost como una cadena de filtros. Cada filtro es un clasificador AdaBoost separado que consta de unos pocos clasificadores débiles. Si cada uno de estos filtros en la región de aceptación de la imagen muestra que el rostro falla, esta área se clasifica inmediatamente como no rostro. Cuando un filtro acepta un área de imagen como

rostro, el área ingresa al siguiente filtro en la cadena. Si esta área de la imagen pasa con éxito todos los filtros de cadena, se clasifica como rostro.

En este algoritmo, cada ciclo de potenciar una característica entre todas las demás características potenciales se selecciona y, al final, la clasificación final será una combinación lineal de la clasificación inicial débil.

*Dado un conjunto de imágenes $(x_1, y_1), \dots, (x_n, y_n)$ donde $y_i = 0, 1$ para muestras negativas y positivas respectivamente.

*Inicializar los pesos $w_{t,i} = \frac{1}{2m} \cdot \frac{1}{2l}$ para $y_i = 0, 1$, donde m es el número de muestras negativas y l es el número de muestras positivas.

*Para $t=1, \dots, T$:

1. Normalizar los pesos

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}} \quad (2)$$

Donde w_t es la distribución de probabilidad

Por cada característica j, entrena un clasificador h_j el cual es restringido por una simple característica. El error es evaluado con respecto a w_t

$$\epsilon_j = \sum_i w_i |h_j(x_i) - y_i| \quad (3)$$

2. Se escoge el clasificador con menor error

3. Se actualizan los pesos

$$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i} \quad (4)$$

Donde $e_i = 0$ si el ejemplo x_i se clasifica correctamente y 1 en caso contrario y

• El clasificador final es

$$h(x) = \begin{cases} 1 & \text{si } \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{en caso contrario} \end{cases} \quad (5)$$

$$\text{donde } \alpha_t = \frac{1}{\beta_t}$$

(4) Clasificador en cascada

En vez de construir un único clasificador mediante el proceso Adaboost, se pueden construir clasificadores más pequeños y eficientes que rechacen muchas ventanas negativas (es decir, aquellas que no incluyan ninguna instancia del objeto buscado) manteniendo casi todas las positivas (es decir, las que contienen una instancia

del objeto buscado). Estos clasificadores más simples se utilizan para rechazar la mayoría de las ventanas de búsqueda y solo en aquellas en las que hay mayores probabilidades de encontrar caras se llama a clasificadores más complejos que disminuyan el número de falsos positivos.



Figura 4 Clasificador en cascada

Se obtiene así una cascada de clasificadores, cada uno de los cuales es entrenado con AdaBoost y después sus valores umbrales se ajustan para minimizar los falsos negativos. La cascada entrenada por Viola-Jones tiene 38 etapas y más de 6000 características, pero de media se evalúan únicamente 10 características por ventana de búsqueda.

b) Detección del rostro

Empleando el algoritmo de Viola-Jones explicado anteriormente, se hace posible un detector de caras robusto basado en un clasificador en cascada.



Figura 5 Detección del rostro

2) Preprocesado de la imagen

En esta segunda etapa, se le aplican una serie de transformaciones a cada rostro detectado anteriormente para que la extracción de características sea lo más efectiva posible. Las operaciones que se llevan a cabo en esta fase son el escalado, el recorte y la ecualización del histograma. Tras finalizar esta fase se obtiene una imagen por cada rostro detectado, en la que aparece únicamente el rostro ecualizado.

El escalado consiste en proporcionar un tamaño

determinado al rostro de forma que todos tengan las mismas proporciones. Para realizar esta operación se aconseja que la distancia entre el centro de los dos ojos. Tomando esa distancia como referencia para el proceso de escalado, se consigue que todas las imágenes tengan unas proporciones muy similares facilitando, así, las tareas de comparación. Después del proceso de escalado, es el momento de recortar la imagen.

Por último, para acabar con el proceso de normalización, se realiza una ecualización del histograma a la imagen recortada. El objetivo de esta transformación es conseguir que el histograma de la imagen convertida a escala de grises sea lo más uniforme posible, es decir, que el número de píxeles para cada nivel de gris sea lo más igualado posible.



Figura 5 Procesamiento de la imagen

Extracción de las características faciales

La extracción de características en imágenes y secuencias de imágenes faciales consiste en extraer información asociada con la activación de los diferentes músculos del rostro, esta tarea puede realizarse en forma global u holística en donde se analiza el rostro como un solo conjunto o localmente en donde se seleccionan regiones de interés del rostro como ojos, cejas y boca.

La extracción de características se emplea para obtener la información que resulta relevante de cara a realizar una comparación. Durante las últimas décadas se han desarrollado un gran número de algoritmos para llevar a cabo dicha extracción en el ámbito del reconocimiento facial. En este trabajo se estudiará la técnica

dependiente de pose Patrones Locales Binarios (LBP), puesto que no requiere un equipamiento especializado para la captura de imágenes o capturas muy específicas controladas.

Algoritmo de Reconocimiento Facial usando Patrones Binarios Locales

El descriptor o algoritmo de Patrones Binarios Locales (LBP) es uno de los más conocidos y ampliamente usado en el contexto de reconocimiento de rostros. La fórmula genérica para calcular una descripción de patrones locales para un pixel p_c dado, es el siguiente:

$$LBP_{p_c} = \sum_{i=0}^7 s(p_i - p_c) 2^i \text{ donde } s(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (6)$$

En la ecuación p_c y p_i , $i=0,1 \dots 7$, serán el valor en escala de grises del pixel central y sus vecinos respectivamente. El operador LBP produce 2^i diferentes valores de salida correspondientes a los 2^i distintos patrones binarios formados por i pixeles de su alrededor. Si la imagen es rotada, estos pixeles a su alrededor de cada vecino se moverán correspondientemente junto al perímetro de la muestra, resultando en un distinto valor de LBP.

En la figura 6, se muestra un ejemplo de la operación básica LBP para el cálculo de un pixel dado. Una operación de umbralización $S()$ transforma los elementos de la matriz construida a partir de pixel central y sus vecinos, en binarios comparándolos con el valor del elemento central. Los números obtenidos son concatenados en dirección de las manecillas del reloj y un nuevo valor de etiqueta para el pixel central es calculado. El operador LBP es usado a menudo para vecindades circulares con diferentes radios. Este descriptor es robusto a cambios de monotonos de iluminación y además relativamente fácil de calcular.

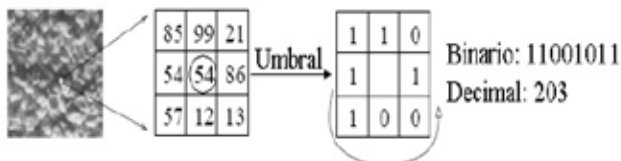


Figura 6 Ejemplo básico de operador LBP

Los histogramas son construidos, por las etiquetas de los pixeles. Una concatenación de todas las descripciones para cada bloque lleva a la obtención de la geometría global del rostro codificada en un solo vector. Las ventajas de tales propuestas radican en su robustez a cambios en la pose, expresión e iluminación.

4) Reconocimiento del rostro

Finalizada la extracción de características se llega a

la última fase, cuyo objetivo es determinar qué imagen del conjunto de entrenamiento es más parecida a la imagen de la base de datos, a partir de sus representaciones mediante las LBP. Para calcular la distancia se usa la distancia euclídea con el fin de tomar una decisión.

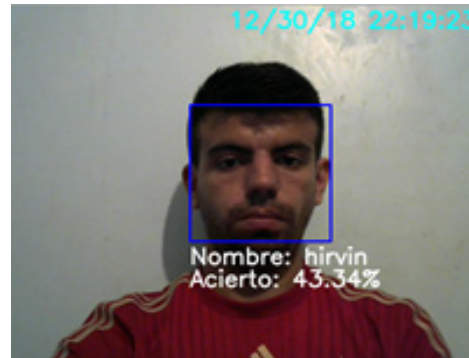


Figura 7 Reconocimiento del usuario

En esta fase se analiza la distancia entre la proyección de la imagen del usuario donde se quiere hacer el reconocimiento y las correspondientes a las imágenes de entrenamiento en la base de datos. El resultado corresponderá con el acierto que se obtuvo en reconocer al usuario con respecto a las imágenes almacenadas.

a) Distancia Euclídea

La distancia euclídea es una de las medidas más básicas para calcular distancias. Esta distancia se define como la distancia directa entre dos puntos en un plano. El ejemplo más claro es la distancia entre dos puntos en un plano de dos dimensiones de coordenadas x y y . Si tuviéramos dos puntos P_1 y P_2 con coordenadas (x_1, y_1) y (x_2, y_2) respectivamente, el cálculo de la distancia euclídea entre los mismos sería

$$d_E(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (7)$$

En general, la distancia euclídea entre dos puntos $P=(P_1, P_2, \dots, P_n)$ y $Q=(Q_1, Q_2, \dots, Q_n)$ en el espacio euclídeo n -dimensional vendría definida

$$d_E(P, Q) = \sqrt{\sum_{i=1}^n (P_i - Q_i)^2} \quad (8)$$

III. RESULTADOS

Se procederá a evaluar el rendimiento que ofrece el sistema de reconocimiento facial. Las pruebas que se llevan a cabo son para comprobar diferentes características del funcionamiento del sistema. Se realizaron diferentes pruebas a 4 usuarios utilizando diferentes características que se puedan presentar utilizando diferentes tipos de condiciones para comprobar en cuál de los casos el sistema presenta los mejores resultados y así

tener un mejor reconocimiento del usuario.



Figura 8 Usuarios registrados

A.Reconocimiento facial con respecto al número de imágenes de entrenamiento

Se realizaron pruebas para comprobar que tan eficiente es el sistema identificando al usuario, se consideraron varios números de imágenes de entrenamiento para ver qué tan eficiente es el sistema en reconocer a diferentes usuarios, para esta prueba se las imágenes tomadas en diferentes ambientes, posiciones, e iluminación. Se realizaron las pruebas para cada uno de los usuarios.

Tabla I.Reconocimiento facial con respecto al número de imágenes de entrenamiento

Nº de imágenes	Usuario 1	Usuario 2	Usuario 3	Usuario 4
20	27,02	52,84	41,51	42,95
30	31,96	56,22	44,99	46,93
50	33,46	62,06	51,26	53,81
75	39,34	65,93	56,24	58,10
100	45,15	69,78	64,79	69,19
150	48,27	72,25	70,82	73,51
200	52,96	75,71	73,92	77,42

Se puede apreciar que a medida que la cantidad de imágenes aumentan, la exactitud que tiene en reconocer al usuario aumenta también, considerando que es necesario una cantidad necesarias de imágenes de entrenamiento para que el sistema obtenga una mayor respuesta. Hay que tener en consideración que a medida que aumenta el número de imágenes de entrenamiento la capacidad de procesarlas es mayor, haciendo que el sistema disminuya la velocidad de procesamiento para reconociendo del usuario.

B.Reconocimiento facial con respecto al tamaño de

las imágenes de entrenamiento.

Para comprobar que tan eficiente es el reconocimiento facial se utilizaron diferentes tamaños para las imágenes de entrenamiento en la base de datos, para este caso se utilizaron 200 imágenes de entrenamiento con las imágenes tomadas en el mismo ambiente, posición e iluminación. Se hicieron las pruebas para cada uno de los usuarios, en los resultados obtenidos que se mostrarán a continuación.

Tabla II.Reconocimiento facial con respecto al tamaño de las imágenes

Tamaño de las imágenes (píxeles)	Usuario 1	Usuario 2	Usuario 3	Usuario 4
Normal	87,45	74,37	50,37	78,28
Normal + 20	80,67	65,76	47,67	67,83
Normal + 40	73,32	61,57	41,26	63,92
Normal + 60	69,08	57,68	35,63	56,90
Normal + 80	62,39	50,96	32,32	50,02
Norma + 100	57,50	44,88	25,51	43,46

Aumentado el tamaño de las imágenes de entrenamiento podemos observar que estas empiezan a disminuir el porcentaje de acierto con respecto a la persona que se este reconocimiento, teniendo así que para poder obtener un resultado con mayor precisión es necesario que la imagen solo sea el rostro del usuario sin que las imágenes del entrenamiento capten el entorno que lo rodean.

C.Reconocimiento facial con respecto a la iluminación

Variando las condiciones de iluminación del ambiente donde se hacían el reconocimiento facial, evaluando que tan eficiente es el sistema cuando se presenta variaciones con respecto a la iluminación para cada uno de los usuarios registrados. Para estas pruebas se utilizaron diferentes locaciones, se contaron con 200 imágenes de entrenamiento para cada uno de los usuarios, con imágenes del mismo tamaño.

Los rangos de iluminación que se presentan son los valores que se pueden encontrar en una oficina o un hogar con una buena iluminación.

Tabla III. Reconocimiento facial con respecto a la iluminación

Iluminación (Lux)	Usuario 1	Usuario 2	Usuario 3	Usuario 4
40	45,39	45,06	50,83	59,17
90	56,17	47,69	54,07	63,65
145	58,31	51,79	60,48	66,07
225	72,12	59,32	69,75	71,28
550	83,35	62,21	76,02	75,73
1000	89,73	76,13	83,64	80,34

Utilizando diferentes condiciones de iluminación de menor a mayor, se pudo ver que la capacidad de reconocer al usuario aumenta progresivamente, teniendo en consideración que la iluminación no interfiera con la lente de la cámara, ya que esto ocasiona que el reconocimiento no se pueda realizar.

IV. CONCLUSIONES

El reconocimiento facial es de gran utilidad para la identificación de personas, ya que es un método no intrusivo y que no requiere de mucha colaboración por parte del usuario al momento del reconocimiento.

Es necesario realizar una base de datos del rostro actualizada del usuario con diferentes ángulos, expresiones y con una cantidad bastante grande de imágenes, para que el reconocimiento facial pueda identificar a la persona de una manera más precisa y eficiente.

Contar con una cámara de una gran resolución colocada en una posición favorable y con una buena iluminación que el reconocimiento del usuario sea más eficaz, con mayor precisión y que no se vea afectado por elementos externos.

Hay diferentes elementos que pueden hacer que el reconocimiento facial no pueda identificar al usuario al momento de su funcionamiento, por lo que es necesario tener este tipo de sistema en un ambiente controlado y también que los usuarios sean consiente de este sistema para que este pueda trabajar lo mejor posible.

V. REFERENCIAS

- [1] Borroso, C. (2014). Implementacion del algoritmo de deteccion de caras de viola y jones sobre un FPGA. Madrid: Universidad Carlos III de Madrid.
- [2] Cabello, E. (2009). Tecnica de reconocimiento facial mediante redes neuronales. Madrid: Universidad Politecnica de Madrid
- [3] Dominguez, S. (2017). Reconocimiento Facial mediante el analisis de componentes principales (PCA). Sevilla: Universidad de Sevilla.
- [4] Espinoza, D., & Jorquera, P. (2015). Reconocimiento

Facial. Valparaiso: Pontificia Universidad Catolica del Valparaiso.

[5] Guevara, M. (2008). Deteccion de rostros en imagenes digitales usando clasificadores cascada. Pereira: Scientia et Technica

[6] Holman, A. (2015). Uso de redes neuronales para el reconocimiento de rostros en ambientes controlados. Bogota: Universidad Distrital Francisco Jose de Caldas

[7] Ramirez, J. (2011). Redes neuronales artificiales para el procesamiento de imagenes,. RIEE&C

[8] Chen, D.-S., & Zhen-Kai, L. (2007). Generalized Haar-Like Features for Fast Face Detection. Hong Kong: International Conference on Machine Learning and Cybernetics.

[9] Viola, P., & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. Kauia: IEEE Computer Society Conference on Computer Vision and Pattern Recognition.

[10] Viola, P., & Jones, M. (2001). Robust Real-Time Face Detection. Vancouver: Second International Workshop on Statistical and Computational Theories of Vision- Modeling, Learning, Computing and Sampling.