DESARROLLO DE UN SISTEMA PROTOTIPO DE ACCESO A LOS LABORATORIOS DE REDES DE LA FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA (FIEE) DE LA ESCUELA POLITÉCNICA NACIONAL (EPN) BASADO EN RECONOCIMIENTO FACIAL

Arroyo Christian, Calderón Xavier

Ingeniería Eléctrica y Electrónica, Escuela Politécnica Nacional

Quito, Ecuador

christian.arroyo@epn.edu.ec

xavier.calderon@epn.edu.ec

***Resumen*- A lo largo del mundo diferentes empresas, compañías e incluso personas comunes hacen uso de la tecnología de reconocimiento facial; esta tecnología permite acceder a diferentes lugares e incluso les permite obtener información personal. Esta misma tecnología es usada habitualmente en la telefonía móvil; ya que muchos celulares actuales proveen de este servicio para desbloquear el dispositivo. Este trabajo propone implementar un sistema prototipo de acceso al Laboratorio de Redes de la Facultad de Ingeniería Eléctrica y Electrónica de la Escuela Politécnica Nacional basado en reconocimiento facial, huella dactilar y PIN de acceso; para lo cual se utiliza el entorno de desarrollo integrado "Visual Studio" con el lenguaje de programación orientado a objetos "C#" para la aplicación de escritorio en conjunto con las librerías OpenCV, EmguCV, Ozeki, ZKTeco e iText sobre un ambiente Windows. Como hardware se utiliza: cámara IP, lector de huella dactilar, panel numérico, router, usb hub y un computador donde se alojará la aplicación de escritorio y la base de datos. El prototipo está orientado a tratar de mejorar el sistema de seguridad ya existente del Laboratorio de Redes de la FIEE de la EPN.**

Palabras clave: Reconocimiento facial, Huella dactilar, PIN de acceso, Biométricos, Visual Studio, OpenCV, EmguCV, Ozeki, ZKTeco e iText.

I. INTRODUCCIÓN

En los últimos años el uso de biométricos, como el reconocimiento facial y huella dactilar, como sistemas de seguridad ha ido creciendo; ya que estos permiten extraer características físicas de las personas como lo son el iris y pupila de los ojos, nariz, cejas, boca, geometría facial, patrón dactilar, etc.

El que exista esta tecnología y que sea aplicable, en varios campos de la vida cotidiana, indica que es posible aplicarlo para incrementar la seguridad del Laboratorio de Redes de la FIEE en la EPN; ya que no cuenta con un sistema de seguridad electrónico para el acceso a las diferentes salas del Laboratorio de Redes.

El prototipo está orientado a tratar de mejorar el sistema de seguridad ya existente del Laboratorio de Redes de la FIEE de la EPN, donde se encuentran los diferentes tipos de equipos que son usados por estudiantes y profesores en sus actividades cotidianas

II. ARQUITECTURA DEL SISTEMA

El sistema prototipo de seguridad del Laboratorio de Redes de la FIEE - EPN constará de un sistema de control de acceso (rostro, huella digital y PIN de acceso), un sistema de interconexión que permita la comunicación de los equipos con los dispositivos intermedios y este a su vez con un computador donde se encuentre alojada la base de datos del sistema de seguridad y la respectiva aplicación desarrollada en C#.

La aplicación estará compuesta de varios módulos, como se muestra en la Fig. 1, los mismos que permitirán realizar las tareas de administración y control del sistema prototipo de seguridad; estos módulos serán:

- Interfaz de Usuario: será el módulo mediante el cual los usuarios podrán interactuar con la aplicación.
- Lógica de Negocios: este módulo realizará la conexión entre el cliente y el servidor, permitirá

reconocer a los dispositivos periféricos y determinar si estos están activos o no para su funcionamiento; adicionalmente, el módulo permitirá recibir los datos enviados por los dispositivos periféricos los cuales serán procesados; además este módulo se comunicará con el módulo de "Acceso a Datos" para solicitar al gestor de "Base de Datos" realizar consultas SQL.

- Acceso a Datos: proporcionará un acceso a los datos guardados en la "Base de Datos"; además permitirá realizar consultas SQL.
- Base de Datos: es donde se encuentran todas las tablas requeridas con toda la información para el correcto funcionamiento de la aplicación de escritorio.

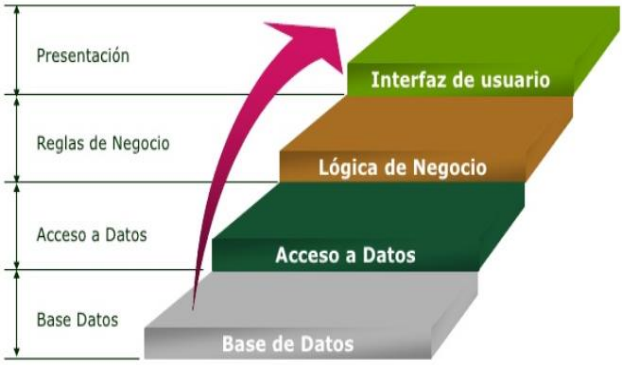


Fig. 1. Módulos del prototipo de sistema de seguridad [1].

III. HERRAMIENTAS SOFTWARE

En esta etapa se mostrará la información necesaria sobre las herramientas software que describa la información relevante sobre las librerías, lenguaje de programación y base de datos utilizadas.

A. Microsoft Visual Studio 2017

Visual Studio es un IDE utilizada por los programadores, ya que proporciona un entorno rico e integrado para crear aplicaciones para Windows, Android e iOS; siempre y cuando su sistema operativo soporte la plataforma .NET; para ello Visual Studio usa plataformas de desarrollo de software de Microsoft como: Windows Forms, API de Windows, Windows Presentation, etc.; adicionalmente, cabe mencionar que Visual Studio provee de un diseñador de clases y un diseñador de esquema de base de datos.

El IDE es una aplicación informática que permite a los desarrolladores varios servicios que facilita el desarrollo de software; básicamente un IDE está compuesto por una serie de librerías y servicios que pueden ser manipuladas a través de una GUI, un editor de código fuente y un depurador; en ciertos casos el IDE puede estar compuesto por un compilador, un intérprete y un auto-completado inteligente de código. "Visual Studio incluye un editor de código compatible con IntelliSense [2]" (auto-completado inteligente de código), así como un depurador.

Visual Studio 2017 ofrece a sus usuarios una serie de mejoras, con respecto a sus predecesores, las mismas que se enfocan al desarrollo de aplicaciones móviles, aplicaciones web, aplicaciones en la nube, desarrollo Azure, entre otras; entre las nuevas características que ofrece se puede mencionar: conjunto de herramientas .NET Core, editor XAML, IntelliSense, pruebas de unidad en vivo, mejora de depuración, etc.

Este es un resumen general de los aspectos más importantes de Visual Studio 2017 [3]:

- Redefinición de los aspectos básicos: el proceso de instalación es más rápido e intuitivo, ya que cuenta con una interfaz más sencilla donde se puede escoger, de una lista, los aspectos que el usuario necesite.
- Rendimiento y productividad: implementa nuevas funciones de desarrollo en la nube y aplicaciones móviles. Adicionalmente, tiene mayor capacidad de respuesta y menor requerimiento en el uso de memoria.
- Desarrollo de aplicaciones en la nube con Azure: permite la creación de aplicaciones destinadas a la nube, facilitando la configuración, la compilación, la depuración, el empaquetado y la implementación de aplicaciones y servicios.
- Desarrollo de aplicaciones para Windows: permite la creación de aplicaciones para una amplia gama de dispositivos Windows como: computadoras con sistema operativo Windows 10, tabletas, teléfonos, Xbox y más.
- Desarrollo multiplataforma: con Visual Studio es posible distribuir software (aplicaciones y librerías) que se ejecuten en sistemas operativos Windows, Linux y macOS sin problemas.

1) ¿Qué es Microsoft .Net Framework?: Para aquellos usuarios de Windows, es probable, que revisando las características del sistema operativo se hayan encontrado con programas instalados de ".NET Framework" y se han preguntado qué funcionalidad tiene ese programa. ".NET es una plataforma de ejecución de Aplicaciones basada en objetos y desarrollada por la empresa Microsoft destinada a facilitar la vida de los desarrolladores de aplicaciones (en especial a los usuarios de Visual Basic y Visual Interdev) [4]". Como ventajas del uso de .NET se puede nombrar [4]:

- La disminución de los tiempos de desarrollo.
- La compatibilidad, portabilidad y reutilización de código entre plataformas operativas y lenguajes de desarrollo.

- La transparencia de ubicación de código.
- Un mejor control de versiones, tanto de la aplicación como de las librerías de clases.

Para el caso de Visual Studio 2017, se detalla en la Tabla I los Frameworks con los que puede trabajar [5].

TABLA I
FRAMEWORKS COMPATIBLES CON VISUAL STUDIO 2017

FRAMEWORKS VISUAL STUDIO 2017
.NET Framework 2.0
.NET Framework 3.0
.NET Framework 3.5
.NET Framework 4.0
.NET Framework 4.5
.NET Framework 4.5.1
.NET Framework 4.5.2
.NET Framework 4.6
.NET Framework 4.6.1
.NET Framework 4.7
.NET Framework 4.7.1
.NET Framework 4.7.2

2) Windows Communication Foundation (WCF): WCF es un marco de Visual Studio para crear aplicaciones orientadas a servicios facilitando la creación de los puntos de conexión, "los puntos de conexión son los lugares donde los mensajes se envían o reciben (o ambos), y definen toda la información necesaria para el intercambio de mensajes [6]"; mediante WCF es posible crear aplicaciones Cliente – Servidor, donde uno de los extremos del servicio actúa como cliente enviando peticiones y el otro extremo actúa como servidor donde las peticiones son procesadas y retorna una respuesta al cliente; los mensajes que se intercambian pueden ser tan simples como un solo carácter, una palabra o incluso secuencia de datos binarios [7].

WCF incluye una serie de características, de las cuales las más importantes son:

- Orientación a servicios: WCF permite crear aplicaciones orientadas a servicios.
- Interoperabilidad: WCF implementa estándares de interoperabilidad del servicio Web.
- Varios patrones de mensajes: Los mensajes se pueden intercambiar de diferentes maneras. El más común es el de solicitud/respuesta. Existen otros tipos de mensajes, como el unidireccional, donde un extremo envía un mensaje sin esperar ninguna respuesta, el mensaje dúplex donde ambos puntos de conexión pueden enviar mensajes al otro de manera independiente.
- Seguridad: Es posible cifrar los mensajes para proteger la privacidad, así como obligar a los usuarios a que se autentiquen antes de permitirles recibir mensajes.
- Se admiten los canales TCP/IP de comunicación.

3) C Sharp (C#): C# es un lenguaje altamente completo; se puede usar C# para crear aplicaciones cliente de Windows, componentes distribuidos, aplicaciones cliente-servidor, aplicaciones de base de datos y más cosas. Visual C# proporciona un editor de código avanzado, prácticos diseñadores de interfaz de usuario, un depurador integrado y otras herramientas que facilitan el desarrollo de aplicaciones basadas en C# y .NET Framework [8].

Una de las fortalezas de C# es que permite el uso de LINQ lo que convierte una consulta fuertemente compleja en una construcción de lenguaje de primera clase; es decir, simplifica y hace más sencillas las consultas a una base de datos.

Los tipos de datos, es algo muy importante a considerar, en .NET se utilizan los mismos tipos de datos para definir los valores de las variables y las constantes; aunque dependiendo del tipo de lenguaje pueden tener nombre diferente. Los tipos de datos, básicos, están en la biblioteca de clases de .NET que se detallan en la Tabla II.

TABLA II
TIPOS DE DATOS, BÁSICOS, EN LA BIBLIOTECA DE CLASES DE .NET [9]

TIPO DE DATO	VB	C#	CONTIENE
Byte	Byte	Byte	Entero de 0 a 255
Int16	Short	Short	Entero de -32.768 a 32.767
Int32	Integer	Int	Entero de -2.147.483.648 a 2.147.483.647
Int64	Long	long	Entero de aproximadamente -9.2e18 a 9.2e18
Single	Single	float	Decimal de precisión simple de aproximadamente -3.48e38 a 3.48e38
Double	Double	double	Decimal de doble precisión de aproximadamente -1.8e308 a 1.8e308
Decimal	Decimal	decimal	Número de 128 bits fraccional de punto fijo que admite hasta 28 dígitos significativos
Char	Char	Char	Un único carácter Unicode de 16 bits
String	String	string	Una serie de caracteres de longitud variable de caracteres Unicode
Boolean	Boolean	bool	Un valor True o False
Object	Object	object	La clase base de todos los tipos .NET, puede contener cualquier tipo de datos u objetos (incluyendo los anteriores)

B. LINQ a SQL

LINQ es un conjunto de herramientas diseñadas para reducir la complejidad de consultas SQL en Visual Studio y Microsoft .Net Framework, evitando así el uso de un lenguaje complicado y reduciendo el tiempo de consulta a la base de datos y los tiempos de proceso de la aplicación como tal. Las consultas LINQ están introducidas dentro del código de la aplicación, lo que permite acceder a las bases de datos de SQL Server, archivos XML e incluso a base de datos de terceros alojadas en sitios distantes.

SQL es un lenguaje de consulta de base de datos normalizado de cuarta generación y utilizado por diferentes motores de base de datos para definir, gestionar y manipular los datos alojados en una base de datos. "Para utilizar SQL desde un lenguaje de programación se necesita sentencias especiales que permitan distinguir entre las instrucciones del lenguaje de programación y las sentencias de SQL [10]", como es el caso de uso de LINQ en el lenguaje de programación de C# en Visual Studio.

LINQ a SQL es una extensión de ORM (Object-Relational Mapping – Mapeador de Objetos Relacionales), el cual permite una conversión de los datos de un objeto a un formato correcto para poder realizar, por ejemplo, funciones básicas CRUD (Create, Read, Update and Delete – Crear, Leer, Actualizar y Borrar) dentro de una base de datos. En la Fig. 2, se muestra cómo se pueden realizar consultas a una base de datos mediante ORM.

C. OpenCV

Las siglas de OpenCV provienen de "Open Source Computer Vision Library", lo que indica que es una librería destinada para el tratamiento de imágenes, ya sean almacenadas o en tiempo real. "OpenCV (…) se publica bajo una licencia BSD y, por lo tanto, es gratis para uso académico y comercial. Tiene interfaces C++, Python y Java y es compatible con Windows, Linux, Mac OS, iOS y Android. OpenCV fue diseñado para la eficiencia computacional y con un fuerte enfoque en las aplicaciones en tiempo real. Escrita en C/C++ optimizado, la biblioteca puede aprovechar el procesamiento multi-core [12]".

La licencia BSD (Berkeley Software Distribution – Distribución de Software Berkeley) es una licencia permisiva, lo que significa que BSD tiene menos restricciones al permitir a los usuarios apropiarse y cobrar (o no hacerlo) [13] tras hacer una distribución, ampliación o modificación del software; en términos generales BSD permite usar código fuente de software libre y transformarlo en software no libre.

La visión computarizada que provee OpenCV, permite tomar una imagen y transformarla en una nueva representación o modificación; una representación indica que se ha reconocido un objeto como un rostro y una modificación se refiere al cambio de una imagen como puede ser la modificación de una imagen a color a una en escala de grises como se aprecia en la Fig. 3.

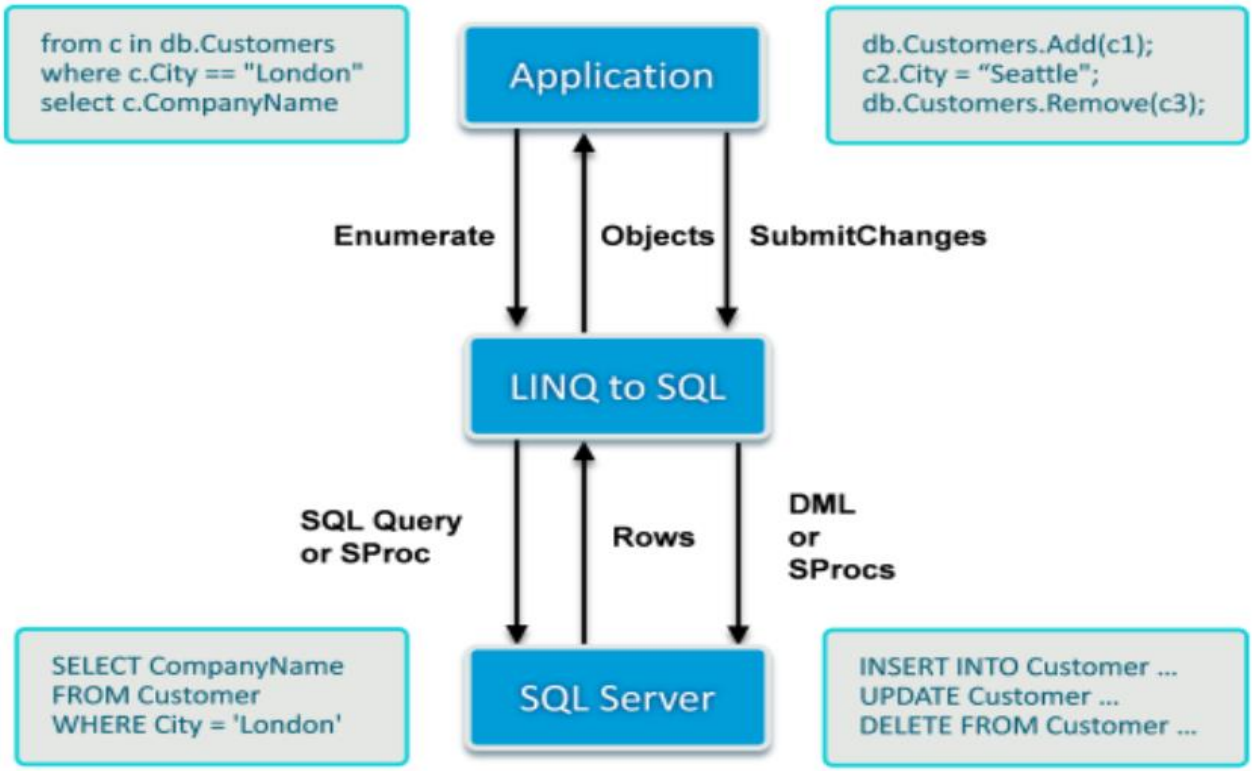


Fig. 2. Mapeador de objetos relacionales [11].

Fig. 3. Representación y modificación de una imagen mediante el uso de OpenCV.

D. EmguCV

EmguCV es un contenedor .Net multiplataforma para la biblioteca de procesamiento de imágenes OpenCV. Permitiendo que las funciones OpenCV sean llamadas desde lenguajes compatibles con .NET como C#, VB, C++, IronPython, etc. El envoltorio puede ser compilado por Visual Studio, Xamarin Studio y Unity, puede ejecutarse en Windows, Linux, Mac OS X, iOS, Android y Windows Phone [14].

Emgu CV está escrito completamente en C#. El beneficio es que puede compilarse en Mono y, por lo tanto, puede ejecutarse en cualquier plataforma compatible con Mono, incluyendo iOS, Android, Windows Phone, Mac OS X y Linux. Otras ventajas del uso de EmguCV son: clase de imagen con color genérico y profundidad, serialización de Imagen XML, la opción de usar la clase de imagen o las funciones de invocación directa de OpenCV, operaciones genéricas en píxeles de imagen, etc.

E. Ozeki SDK

Ozeki Camera SDK es un excelente kit de desarrollo de software para desarrolladores de C# .NET que le permite implementar aplicaciones de cámaras IP y cámaras WEB para sistemas de monitoreo en red, sistemas analíticos de video en red, detectores de movimiento, entre otros. Adicionalmente puede manejar cámaras USB (Universal Serial Bus – Bus Universal en Serie) y RTSP (Real Time Streaming Protocol – Protocolo de Transmisión en Tiempo Real) [15].

El SDK de la cámara OZEKI se basa en los estándares de ONVIF. "ONVIF es un foro líder y reconocido en la industria cuya misión es proporcionar y promover interfaces estandarizadas para la interoperabilidad efectiva de los productos de seguridad física basados en IP [16]". Gracias a la flexibilidad de ONVIF ha hecho que esta tecnología se haga más popular. Los beneficios de ONVIF son:

- Interoperabilidad: permite la comunicación entre los productos de diferentes fabricantes.
- Flexibilidad: se puede trabajar con los productos de varios fabricantes.
- Calidad: el producto está constantemente actualizando y corrigiendo los errores que surgen.

IV. BIOMÉTRICOS

"La biometría es un método de reconocimiento de personas basado en sus características fisiológicas o de comportamiento. Se trata de un proceso similar al que habitualmente realiza el ser humano reconociendo e identificando a sus congéneres por su aspecto físico, su voz, su forma de andar, etc. [17]".

Dentro del campo de los sistemas informáticos, la biometría informática es la aplicación de técnicas matemáticas sobre los individuos ya sea a sus características físicas (estáticas) o a su comportamiento (dinámicas) para lograr una autenticación; para ello se pueden usar diferentes métodos como son: el iris del ojo, las huellas dactilares, geometría de la mano, el rostro de la persona, etc., para el caso de características físicas y para el caso de comportamiento se puede usar el reconocimiento por voz o por firma, entre otros.

Todo sistema biométrico tiene un índice de aceptación o umbral dictado por la Teoría de Reconocimiento de Patrones; esto indica nada más y nada menos el rendimiento que tiene el sistema biométrico. Si el umbral del sistema biométrico es demasiado bajo, existe la posibilidad de que personas no autorizadas puedan ingresar al sitio custodiado por el sistema biométrico y si por el contrario el umbral es demasiado alto puede suscitarse de que los usuarios autorizados del sistema no puedan ingresar; por lo cual es necesario establecer un umbral adecuado al método escogido del sistema biométrico. Los sistemas biométricos actuales tienen un rendimiento que va desde el 60% para los más bajos hasta el 99.9% para los más exigentes [18].

El rendimiento de un sistema biométrico está dado en los siguientes términos:

- Tasa de Falsa Aceptación (FAR): indica cual es la tasa de que personas no autorizadas puedan ingresar.
- Tasa de Falso Rechazo (FRR): también conocida como FNMR, indica la tasa de rechazo a persona autorizada.
- Tasa de Error de Reclutamiento (FTE): también conocida como FER, indica el porcentaje de usuarios que no pueden registrarse en el sistema, esto es debido al método biométrico utilizado (estático o

dinámico); ya que según el método se deberían tomar varias muestras del patrón del individuo, este es el caso para sistemas basados en voz, donde se debe considerar la pronunciación que realiza el individuo.

- Tasa de Error Igual (EER): indica la tasa donde el FAR y el FRR son iguales.
- Tasa de Error de Cruce (CER): indica la sensibilidad del sistema biométrico, es el punto donde se cruza el FAR y el FRR.

Los valores FAR y FRR pueden ser observados de manera más detallada en la Fig. 4, donde se detalla una gráfica de compensación de error (DET). La curva de rendimiento es calculada usando los resultados después de la comparación con la base de datos, los datos son mostrados en una escala de desviación normal. En la Fig. 5, se muestra el gráfico donde la Tasa de Falsa Aceptación y la Tasa de Falso Rechazo se cruzan; esto indica la sensibilidad del sistema biométrico y por ende indica el índice de error que el mismo posee.

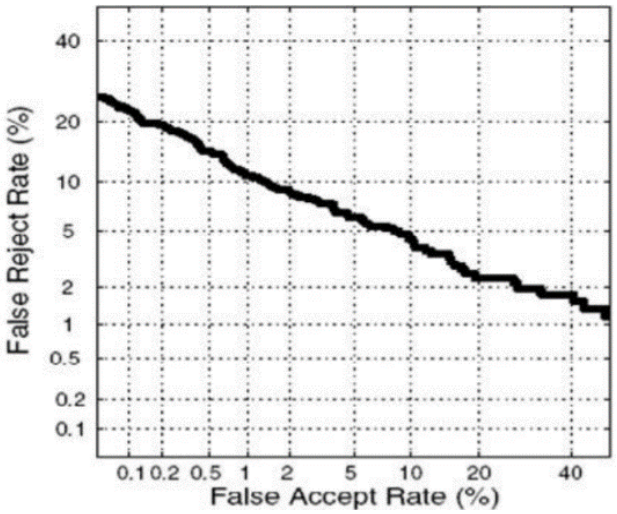


Fig. 4. Rendimiento de un sistema biométrico resumido en un gráfico DET [19].

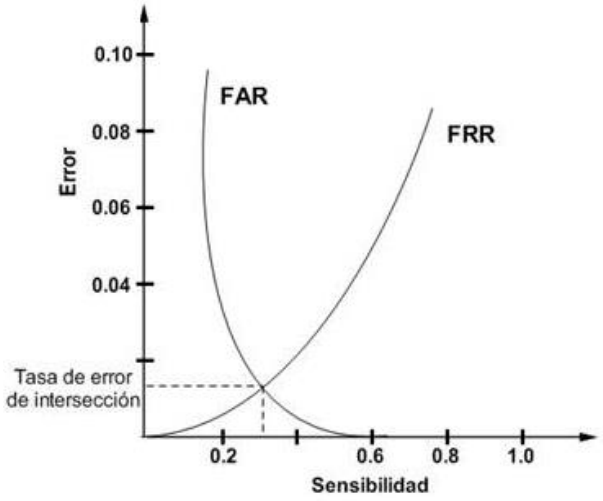


Fig. 5. Sensibilidad del sistema biométrico.

Los sistemas biométricos no son perfectos, aunque están cerca, por lo cual proporcionan ciertas ventajas y desventajas al usuario final que ocupe un determinado sistema biométrico.

Ventajas:

- No es necesario andar a cargar tarjetas o credenciales para el ingreso a un determinado sitio.
- Las tecnologías de acceso mediante sistemas biométricos son más seguras que las habituales como el usuario y contraseña, las cuales deben ser recordados por el individuo y de llegar a olvidarse se deben pasar por ciertos protocolos para poder recuperarlos.
- Es más barato el mantenimiento.
- Las características físicas de una persona no pueden ser trasmitidas a otra persona.

Desventajas:

- Robo de identidad: dado el caso que el sistema biométrico sea vulnerado, los patrones biométricos no pueden ser cambiados. Por ejemplo: si una cuenta de correo es vulnerada, el sistema provee la opción de cambiar la contraseña; no es así en los sistemas biométricos ya que el patrón biométrico es único y no se lo puede modificar.
- Privacidad: al tener un patrón único, esto provee a entidades gubernamentales un fácil rastreo de un individuo y que posiblemente los datos privados sean vigilados u observados.

En la Tabla III, se puede observar las ventajas y desventajas de un sistema biométrico, según sea el método escogido. No se consideran todos los métodos, solo los más usados.

V. FUNCIONAMIENTO DEL SISTEMA

Dependiendo de la tecnología biométrica utilizada, estática o dinámica, se extrae un patrón único para cada persona como la voz o patrón de la cara, para posteriores comparaciones con un patrón previamente almacenado en una base de datos. Para garantizar que el patrón obtenido sea único, el sistema biométrico debe cumplir con los siguientes requisitos:

- Universalidad: todos los individuos deben poseer la misma característica a medir.
- Univocidad: la característica obtenida del individuo lo debe distinguir.
- Permanencia: la característica del individuo debe ser la misma en cualquier momento y lugar.
- Cuantificación: la característica del individuo debe poder medirse.

TABLA III
COMPARACIÓN ENTRE ALGUNOS MÉTODOS BIOMÉTRICOS [20]

	Ojo (Iris)	Huellas dactilares	Geometría de la mano	Escritura y firma	Voz	Cara 2D
Fiabilidad	Muy alta	Muy Alta	Alta	Media	Alta	Media
Facilidad de uso	Media	Alta	Alta	Alta	Alta	Alta
Prevención de ataques	Muy alta	Alta	Alta	Media	Media	Media
Aceptación	Media	Alta	Alta	Muy Alta	Alta	Muy alta
Estabilidad	Alta	Alta	Media	Baja	Media	Media

Los métodos para la identificación de un individuo son muy diversos; pero estos métodos, en esencia, poseen las mismas etapas para el reconocimiento del individuo. Las etapas que se usan para la identificación se pueden reducir a dos etapas, como se muestra en la Fig. 6.

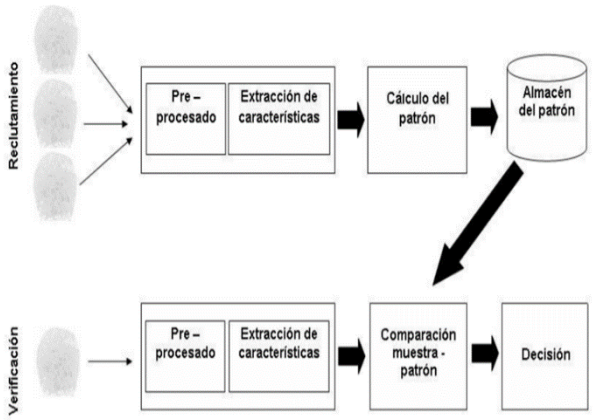


Fig. 6. Etapas en un sistema de identificación biométrica [19].

La etapa de Reclutamiento o Inscripción básicamente es la encargada de tomar las muestras del individuo, las mismas que son procesadas y se obtiene un patrón único sobre el individuo; finalmente, este patrón es almacenado en una base de datos para una futura comparación. Si existen varias muestras aceptables del patrón se realiza un cálculo de la media del patrón, obteniendo así el patrón final a ser almacenado.

En la etapa de Verificación, al igual que en la etapa de inscripción, se toman muestras del individuo y son procesadas, obteniendo un patrón; este patrón finalmente, será comparado con uno almacenado en la base de datos, obteniendo un resultado positivo o negativo.

Cada etapa está conformada por varias sub-etapas las mismas que tienen un propósito en específico, estas son:

- Captura (En la Fig. 6 representado con la huella dactilar): aunque no se muestra directamente en la Fig. 6, esta hace referencia al elemento tecnológico para obtener las características del individuo; es decir, este elemento corresponde a una cámara de video, un lector dactilar, etc.
- Pre-procesado: esta sub-etapa se encarga de adecuar la información obtenida para facilitar su tratamiento posterior. Por ejemplo, aquí es donde se delimitan los bordes de una imagen, se la amplia o reduce, etc.
- Extracción de características: la sub-etapa más importante; ya que aquí es donde son extraídas las características más significativas, el patrón, del individuo. Este patrón es almacenado en forma de plantilla en una base de datos. Es en esta sub-etapa donde se fundamenta la capacidad del sistema biométrico en reconocer entre individuos.
- Comparación: se realizan las tres sub-etapas anteriormente mencionadas obteniendo un patrón del individuo; este patrón es comparado con el almacenado en la base de datos. Esta comparación no se trata de una comparación binaria o de igualdad; más bien se trata de una comparación de variaciones o diferencias entre el patrón obtenido y el almacenado. Así para determinar una decisión positiva o negativa se debe crear un umbral de aceptación, esta aceptación está basada en la Teoría de Reconocimiento de Patrones: Distancia Euclídea, Distancia de Mahalanobis, Distancia de Hamming, Estadísticas utilizando funciones de distribución, clasificadores bayesianos, técnicas basadas en modelado de problemas como Redes Neuronales y Modelos de Mezclas de Gaussianas.

Para la etapa de verificación cabe aclarar que existen dos procesos diferentes para hacerlo, estos son: Identificación y Autenticación. La identificación es tomar el patrón obtenido y hacer una comparación con una serie de patrones almacenados en la base de datos, es decir una comparación uno a varios, donde el resultado de las distintas comparaciones resulta en la identificación del individuo; mientras que la autenticación es donde se toma el patrón obtenido y se lo compara con uno ya guardado en

la base de datos, es decir se hace una comparación uno a uno. Esto se muestra en la Fig. 7.

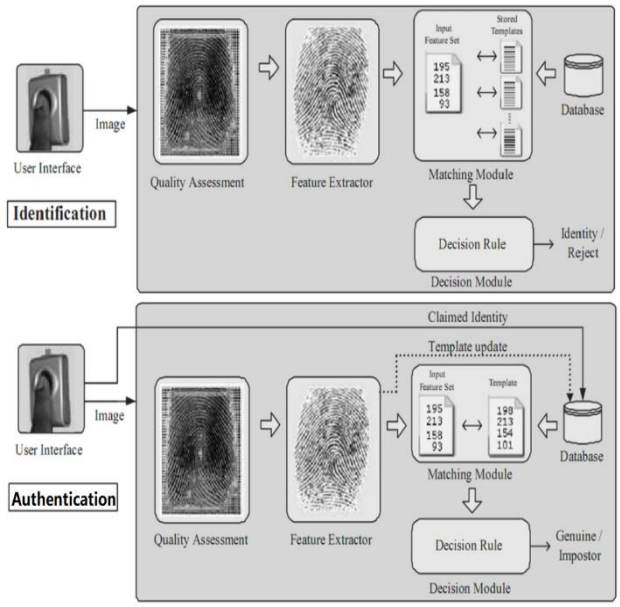


Fig. 7. Identificación (arriba) y autenticación (abajo), diferencias entre las fases [19].

A. Reconocimiento Facial

Se debe distinguir, con precisión, lo que es Detección Facial de lo que es Reconocimiento Facial ya que el reconocimiento facial se basa en la detección facial. Para un sistema biométrico que se base en esta tecnología primero se debe detectar el rostro de una persona y luego hacer el reconocimiento del rostro detectado en una base de datos.

La detección facial es una tecnología que permite determinar el lugar y el tamaño de rostros humanos en imágenes o videos; esto viene a ser un caso particular de la detección de objetos. Si bien para un ser humano, realizar la tarea de ubicar un objeto es relativamente sencillo, no lo es así para una computadora; ya que esta depende de varios factores como lo son: la posición del rostro, iluminación, expresiones faciales, oclusiones (lentes, gorros, parches, etc.), detalles faciales (bigote, cicatrices, etc.).

El algoritmo de Viola – Jones, es el algoritmo para detectar rostros con mayor índice de acierto [21], con un 99.9% de acierto; este algoritmo es muy utilizado ya que no requiere de altos procesos computacionales. Está constituido por tres partes: el primero es la imagen integral, el segundo es el clasificador en cascada, que garantiza una discriminación rápida al descartar objetos de fondo que no son de interés para el análisis; y el tercero es el entrenador de clasificadores, que está basado en Adaboost, el cual es un meta-algoritmo de aprendizaje automático, que garantiza seleccionar las características más importantes de todo el conjunto [22]. Gracias a su potencia y velocidad este algoritmo es implementado en la librería de OpenCV.

El procesamiento que realiza lo hace en imágenes en escala de grises, la misma que es transformada en una imagen integral, lo que significa que se realiza una representación intermedia de la imagen mediante una Tabla de Área Sumada (Summed Area Table) y a su vez permite un cómputo más rápido. En la Fig. 8, se puede observar la imagen original en representación de grises numérico y a la derecha se puede observar la imagen integral.

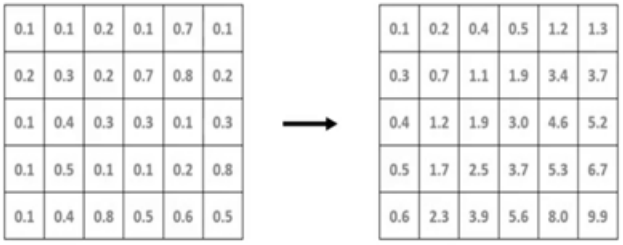


Fig. 8. Cambio de imagen original a una imagen integral.

La imagen integral es una matriz de igual tamaño que la original, esta matriz resulta de la suma de los elementos, pixeles en una imagen, arriba y a la izquierda del punto "X" seleccionado.

El clasificador en cascada (haar), es un conjunto de descriptores que permiten obtener información de un área en particular mediante operaciones matemáticas, esto se lo realiza aplicando filtros haar en el algoritmo de Viola – Jones, estos filtros pueden ser de tres tipos, como se puede observar en la Fig. 9:

- Característica de dos rectángulos: es la diferencia entre la suma de los pixeles de ambas regiones rectangulares.
- Característica de tres rectángulos: Es la suma de los pixeles de los rectángulos exteriores menos la suma del rectángulo central.
- Característica de cuatro rectángulos: Es la diferencia entre los pares diagonales de los rectángulos.

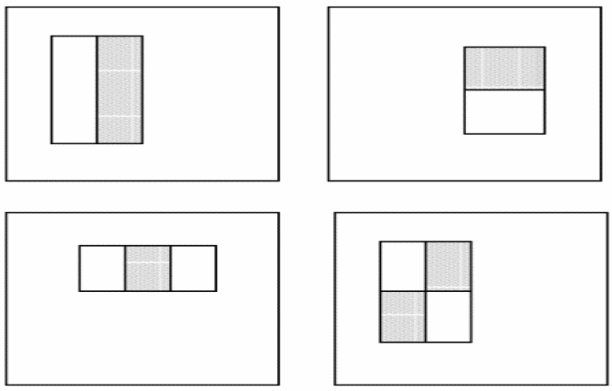


Fig. 9. Características haar [21].

Para determinar si en una imagen se encuentra un rostro o no, el algoritmo de Viola – Jones en conjunto con Adaboost divide la imagen integral en subregiones de diferentes tamaños, donde se utiliza un clasificador en cascada determinando si en cada subregión existe o no un rostro. Este algoritmo funciona mejor con rostros frontales; ya que en rostros con vista de perfil aportan variaciones a la plantilla que no puede manejar.

Las características elegidas por AdaBoost son significativas y de fácil interpretación. La elección de la primera característica se basa en la propiedad que la región de los ojos es más oscura que la región de las mejillas. La segunda característica se basa en que los ojos son más oscuros que la zona de la nariz [23]. Se puede observar lo mencionado en la Fig. 10.

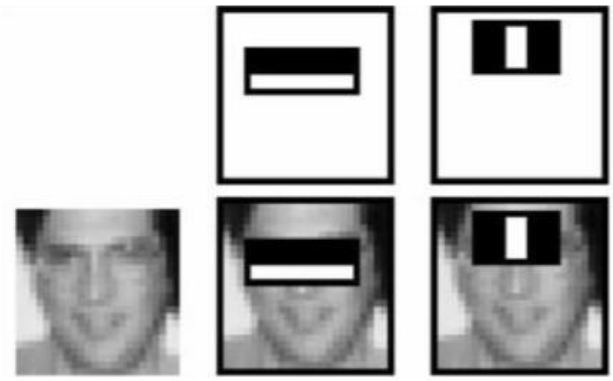


Fig. 10. Las dos características de haar más significativas seleccionadas por AdaBoost [21].

Un sistema biométrico basado en reconocimiento facial, es una aplicación dirigida por un ordenador, que se encarga de identificar a una persona automáticamente de una imagen digital; esta imagen puede ser obtenida por una cámara de video en tiempo real o una que esté almacenada en el ordenador. El ordenador se encarga de hacer una comparación entre la imagen obtenida y una almacenada en una base de datos [24].

Para lograr el reconocimiento facial de un individuo, es necesario de cuatro pasos:

- Detección facial: mediante el uso de un algoritmo, como el de Viola – Jones, se procede a detectar la existencia de un rostro en una imagen o video.
- Alineación facial: localiza los rasgos del rostro en la imagen; esto significa que se procede a ubicar los parámetros característicos del rostro como la boca, la nariz o ceja, así como las distancias entre estas. Aquí también se elige el tamaño de la imagen, así como la gama de colores.
- Extracción de las características faciales: proporciona la información necesaria del rostro detectado para distinguirlo entre personas diferentes.
- Reconocimiento: el patrón extraído en los pasos anteriores se procede con la comparación con uno almacenado en una base de datos; para ello se usa un algoritmo de reconocimiento facial.

En la Fig. 11, se puede observar a detalle el procedimiento realizado para realizar el reconocimiento facial de una persona.

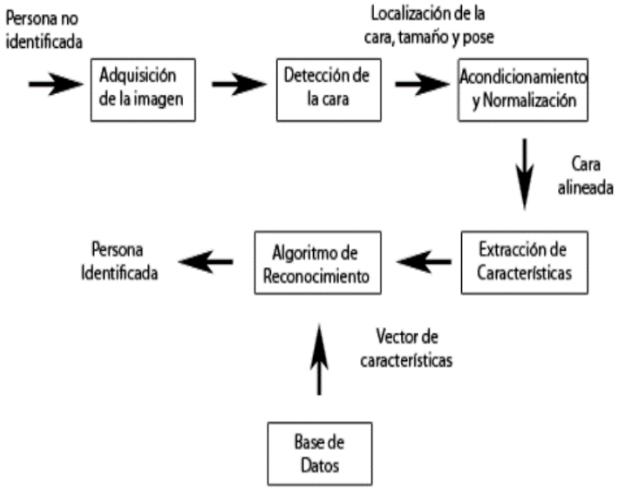


Fig. 11. Procedimiento para el reconocimiento facial [25].

La librería OpenCV cuenta con clases (Facerecognizer) que facilitan el trabajo a la hora de hacer reconocimiento facial, la clase facerecongnizer implementa tres algoritmos: EigenFaces, FisherFaces y LBHP, se puede implementar el que mejor se ajuste a nuestras necesidades.

El algoritmo EigenFaces, es un algoritmo que toma en consideración las características comunes de las personas como son: la nariz, boca, cejas, ojos y las distancias entre ellos, estos componentes en común son llamados EigenFaces.

El algoritmo FisherFaces tiene en cuenta cómo se refleja la luz y las expresiones faciales en el rostro de una persona. FisherFaces clasifica y reduce la dimensión de las caras utilizando el método Discriminante Lineal de Fisher (FLD) y PCA (conocido como EigenFaces). Este método crea una proyección lineal que maximiza las diferentes imágenes de caras proyectadas; pudiendo así hacer comparaciones posteriores mediante el uso de la distancia euclidiana.

El algoritmo LBHP (Local Binary Patterns Histograms – Histogramas de Patrones Binarios Locales) se basa en el algoritmo LBP; ya que se ha determinado que cuando el LBP se combina con los histogramas del descriptor de gradientes, mejora considerablemente el rendimiento de detección.

Usando el LBP combinado con histogramas se puede representar las imágenes de la cara con un simple vector de datos, lo cual facilita el reconocimiento de rostros [26].

Al igual que EigenFaces, LBP necesita de una serie de imágenes conocidas como Imágenes de Entrenamiento, con las cuales el algoritmo podrá reconocer una imagen de entrada y darle una salida. Luego de tener las Imágenes de Entrenamiento, se debe aplicar el algoritmo LBP a cada imagen; finalmente, se debe obtener la imagen con la cual se desea comparar; esta imagen debe estar en escala de grises o ser transformada como tal.

B. Huella Dactilar

El método de huella dactilar para sistemas biométricos, es uno de los más estudiados y uno de los más implementados a nivel mundial; ya que posee una fiabilidad muy alta, es fácil de usar y tiene una amplia aceptación por parte de los usuarios, debido a que existen numerosos estudios que avalan la unicidad de la huella dactilar de los individuos, sin olvidar que ésta no cambia con la edad de las personas.

Las huellas dactilares están constituidas por rugosidades de la piel que forman salientes y depresiones [24]. Para la extracción de las huellas dactilares se debe tener en consideración las características de las huellas dactilares, las más importantes son:

- Minucias: es el punto de interés de la huella dactilar, estos puntos pueden ser utilizados para el reconocimiento dactilar de una persona [27].
- Crestas (Rides): son la parte más sobresaliente, o elevada, de la huella dactilar. Cuando se imprime la huella dactilar las crestas vienen a ser las zonas o rayas negras.
- Valles (Valleys): son la parte profunda de la huella dactilar, esta se encuentra entre los valles de la huella dactilar. Cuando se imprime la huella dactilar los valles vienen a ser las zonas blancas.
- Curvas (Loops): es el sector donde una cresta toma una curva, generalmente en forma de "U". Se lo conoce, también, como lazo.
- Bifurcaciones (Deltas): es donde la cresta en cualquier parte de su recorrido se divide en dos crestas que continúan paralelamente. También es la parte de la huella dactilar donde hay una triangulación.
- Espirales (Whorls): como el nombre lo indica, es una cresta tipo curva, forma una espiral.
- Núcleo (Core): es el centro de la huella dactilar, es donde se genera el inicio del lazo en la huella o el inicio de una espiral.
- Terminación (Ending): es donde la cresta termina y no continua. Esta se ubica entre dos crestas.

En la Fig. 12, se puede observar algunas de las características de las huellas dactilares como son: los valles, las crestas, los lazos, las bifurcaciones o deltas, el espiral y el núcleo de la huella dactilar.

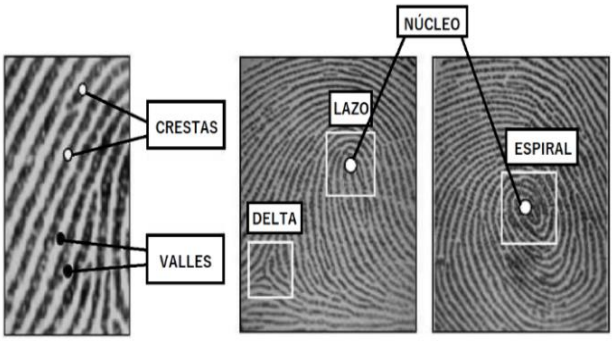


Fig. 12. Algunas de las características de una huella dactilar [19].

Todas las características antes nombradas conforman el patrón de la huella dactilar, este patrón es único incluso entre gemelos idénticos; se estima que la probabilidad de que dos personas tengan la misma huella dactilar es de 1 en 64.000 millones

Para realizar un reconocimiento de una huella dactilar, primero, se debe digitalizar la huella; esto permite obtener las características más relevantes, las minucias, de una huella. Para digitalizar una huella se hace uso de algoritmos que permiten obtener un índice numérico correspondiente a la huella dactilar; cuando una persona desea ser identificada, solo debe poner su dedo sobre el lector dactilar, el mismo que escanea la huella para luego ser comparada con una almacenada en una base de datos.

Las técnicas más utilizadas para la extracción de las huellas dactilares son:

- Segmentación: la segmentación es una tarea de procesamiento de imágenes la cual consiste en separar el área de la huella dactilar del fondo.
- Mejoramiento y Binarización: el objetivo de esta técnica es mejorar la claridad de la estructura de las crestas.
- Técnicas Basadas en Correlación: mediante la utilización de esta técnica se analiza el patrón global seguido por la huella dactilar, es decir, el esquema general del conjunto de la huella en lugar de las minucias.
- Basada en minucias: esta técnica basa su mecanismo de autenticación en determinadas formas fácilmente identificables existentes en la huella dactilar. Así, se registra el tipo de minucia y su posición dentro de la huella, estableciendo una serie de mediciones. De esta

forma, el modelo o plantilla correspondiente a cada usuario es un esquema en el que se indican las minucias que se han de detectar, su posición y las distancias que separan unas de otras [17].

Para poder reconocer a una persona mediante su huella dactilar, se debe proceder a extraer las minucias de una huella de entrada; una vez extraídas las minucias éstas son realzadas, luego son medidas en ubicación y distancia entre ellas. Finalmente, es trasformada en una plantilla la cual es almacenada o comparada con una ya existente en una base de datos. Se puede observar de mejor manera en la Fig. 13.

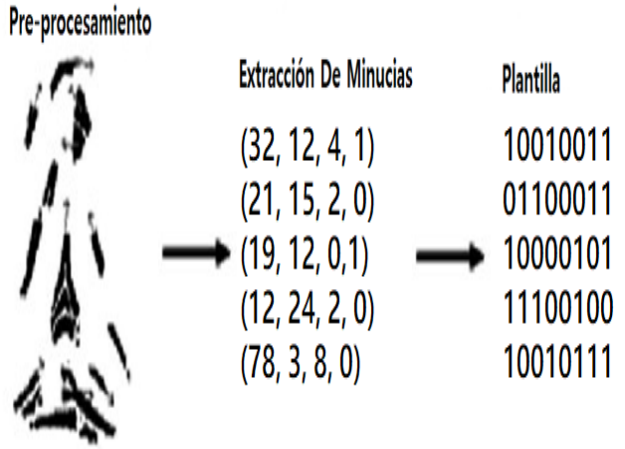


Fig. 13. Procesamiento de una huella dactilar [24].

C. PIN de Acceso

El PIN de acceso o Número de Identificación Personal es utilizado en muchos sistemas electrónicos como pueden ser computadoras, teléfonos celulares entre otros, para poder obtener acceso al sistema o a un lugar restringido. El PIN no viene a ser más que una contraseña numérica que solo el usuario conoce; esta contraseña, generalmente, es un código de 4 dígitos (entre el 0000 y el 9999) como en los cajeros automáticos.

El PIN de acceso es uno de los sistemas que mayor rendimiento tiene ya que el usuario solo debe recordar una secuencia de números corta y su ingreso se lo realiza por un panel numérico. Normalmente los sistemas que trabajan con PIN de acceso no tienen problemas al conceder el acceso ya que solo se requiere de comparar el código ingresado con uno almacenado en la base de datos.

Si bien el PIN de acceso es el que posee mayor rendimiento, es el menos seguro ya que este puede ser obtenido por otras personas, dándole así acceso al sistema o infraestructura a usuarios desconocidos; por lo cual se acostumbra acompañar al PIN de acceso con el uso de tarjetas inteligentes o de banda magnética para una segunda identificación [28].

IV. CONCLUSIONES

El uso de librerías de código abierto como OpenCV y EmguCV es muy útil ya que no solo permiten la implementación de sistemas basados en la fisiología de las personas; sino que también permiten la implementación de detección de movimiento, reconocimiento de objetos, entre otros.

Los sistemas biométricos, son una de las tecnologías más usadas a nivel mundial, estas proporcionan mayor seguridad a los usuarios que las seguridades típicas como contraseñas, patrones de desbloqueo, pin de acceso, etc.; ya que las características físicas de una persona son difíciles de clonar.

La fiabilidad de las huellas dactilares está entre las más altas, los errores producidos con este método son principalmente por no tomar una adecuada muestra de la huella durante el registro del usuario o durante el enrolamiento del usuario (muestra de huella para el acceso).

Durante el estudio de las técnicas para detección y reconocimiento facial y dactilar, se pudo entender de la gran complejidad que poseen los algoritmos biométricos; estos algoritmos están compuestos por altos conceptos y fórmulas matemáticas que combinadas con matrices ayudan a mejorar la seguridad de las personas.

RECONOCIMIENTOS

Los autores agradecen a la Escuela Politécnica Nacional y, especialmente, a la Coordinación de Ingeniería en Electrónica y Redes de Información por el apoyo prestado para el desarrollo de este proyecto.

REFERENCIAS

[1] SlideShare. (2019, septiembre 16). “Diseñando Sistemas empleando el modelo de capas en desarrollo de software”. Por: Ernesto Alexander Calderón Peraza. Docente del Departamento de Ingeniería. Área de Informática [Online]. Disponible en: https://es.slideshare.net/calderonperaza/disenando-sistemas-empleando-el-modelo-de-capas-en-desarrollo-de-software

[2] Visual Studio. (2019, julio 2019). “Aprendizaje Visual Studio 2013” [Online]. Disponible en: https://riptutorial.com/Download/visual-studio-2013-es.pdf

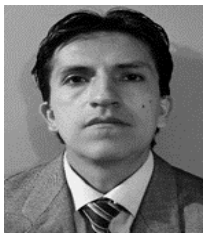
[3] Microsoft Docs. (2019, septiembre 16). “Novedades de Visual Studio 2017” [Online]. Disponible en: https://docs.microsoft.com/es-es/visualstudio/ide/whats-new-visual-studio-2017?view=vs-2017

[4] Federico G. Rudolph, “Introducción a Visual Studio. NET”. 1st edición. Manual de Referencia, Curso de Capacitación en .NET. 2010. Páginas 1 – 6. [Online]. Disponible en: https://es.calameo.com/read/005274837376a635a826b

[5] Microsoft Docs. (2019, septiembre 16). “Compatibilidad y destinatarios de la plataforma Visual Studio 2017” [Online].

Disponible: https://docs.microsoft.com/es-es/visualstudio/productinfo/vs2017-compatibility-vs
[6] Microsoft Docs. (2019, septiembre 16). "Conceptos básicos de Windows Communication Foundation" [Online]. Disponible en: https://docs.microsoft.com/es-es/dotnet/framework/wcf/fundamental-concepts
[7] Microsoft Docs. (2019, septiembre 16). "¿Qué es Windows Communication Foundation?" [Online]. Disponible en: https://docs.microsoft.com/es-es/dotnet/framework/wcf/whats-wcf
[8] Microsoft Docs. (2019, septiembre 16). "Introducción al lenguaje C# y .NET Framework" [Online]. Disponible en: https://docs.microsoft.com/es-es/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework
[9] Microsoft Docs. (2019, septiembre 16). "Resumen de tipos de datos (Visual Basic)" [Online]. Disponible en: https://docs.microsoft.com/es-es/dotnet/visual-basic/language-reference/data-types/
[10] R. Capms, L. Casillas, D. Costal, M. Gibert, C. Martín, O. Pérez, "Base de Datos", 1st edición. Eureca Media, SL. Mayo 2005. Páginas 113 – 118. [Online]. Disponible en: http://www.sw-computacion.f2s.com/Linux/007-Bases_de_datos.pdf
[11] CodeProject. (2019, julio19). "Linq DLinq XLinq PLinq All at one place" [Online]. Disponible en: https://www.codeproject.com/Articles/105098/Linq-DLinq-XLinq-PLinq-All-at-one-place
[12] OpenCV. (2019, septiembre 16). "OpenCV" [Online]. Disponible en: https://opencv.org/
[13] GNU Operating System. (2019, septiembre 16). "El problema de la licencia BSD" [Online]. Disponible en: https://www.gnu.org/licenses/bsd.html
[14] EmguCV. (2019, septiembre 16). "EmguCV" [Online]. Disponible en: http://www.emgu.com/wiki/index.php/Main_Page
[15] Ozeki Camera SDK. (2019, septiembre 16). "Ozeki Camera SDK - Product Guide" [Online]. Disponible en: http://www.camera-sdk.com/p_12-quick-start-guide-for-the-ozeki-camera-sdk-onvif.html
[16] ONVIF. (2019, septiembre 16). "Onvif Organization" [Online]. Disponible en: https://www.onvif.org/about/organization/
[17] Anónimo, "Tecnologías biométricas aplicadas a la ciberseguridad". Revista del Instituto Nacional de Ciberseguridad. Vol. 01. No. 06. 2016, Páginas 4 – 30. [Online]. Disponible en: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_tecnologias_biometricas_aplicadas_ciberseguridad_metad.pdf
[18] Equifer. (2019, septiembre 16). "Lectores Biométricos" [Online]. Disponible en: http://www.equifer.com/lectores-biometricos/
[19] M. Ruiz, J. Rodríguez, J. Olivares, "A glance to the biometric". Revista Avances en Sistemas Informáticos" Vol. 06, No 02, febrero 2009. Páginas 1 – 10. [Online]. Disponible en: http://www.bdigital.unal.edu.co/23395/1/20295-68748-1-PB.pdf
[20] SAD UT3. (2019, juli19). "Biometría" [Online]. Disponible en: http://dis.um.es/~lopezquesada/documentos/IES_1213/SAD/curso/UT3/ActividadesAlumnos/2/html/biometria.html
[21] P. Viola, M. Jones, "Rapid Object Detection using a Boosted Cascade as Simple Features". Computer vision and pattern recognition. 2001. Páginas 1 – 9. [Online]. Disponible en: https://docplayer.es/26824183-Reconocimiento-facial.html
[22] A. Pătraşcu, "Aplicación para Detección y Reconocimiento Facial en Interiores", Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Sevilla – España, 2016. [Online]. Disponible en: http://bibing.us.es/proyectos/abreproy/90722/fichero/ResumenTFG_PatrascuVioricaAndreea.pdf
[23] D. Espinoza, P. Jorquera, "Reconocimiento Facial", Pontificia Universidad Católica de Valparaíso, 2015. [Online]. Disponible en: http://opac.pucv.cl/pucv_txt/txt-1000/UCD1453_01.pdf
[24] C. Tolosa, A. Giz, "Sistemas Biométricos". Anónimo. Anónimo. Páginas 17 – 19 y Páginas 22 – 23. [Online]. Disponible en: https://www.dsi.uclm.es/personal/MiguelFGraciani/mikicurri/Docencia/Bioinformatica/web_BIO/Documentacion/Trabajos/Biometria/Trabajo%20Biometria.pdf
[25] M. Paredes, M. Valle, G. Alvarón, F. Huincho, K. Gutiérrez, "Sistema de vigilancia biométrico para el control delincuencial en la división policial". Revista Conocimiento para el Desarrollo. 2017, Paginas 1 – 7. [Online]. Disponible en: http://repositorio.usanpedro.edu.pe/bitstream/handle/USANPEDRO/293/PI1610091.PDF?sequence=1&isAllowed=y
[26] Medium. (2019, septiembre 16). "Reconocimiento facial: Entendiendo el algoritmo LBPH" [Online]. Disponible en: https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b
[27] A. Rosales, "Clasificación de Huellas Digitales Mediante Minucias". Instituto Nacional de Astrofísica, Óptica y Electrónica, abril 2009. Páginas 1 – 9. [Online]. Disponible en: https://ccc.inaoep.mx/~esucar/Clases-mgp/Proyectos/reporte_modelos_huellas.pdf
[28] Investopedia. (2019, septiembre 16). "Número de Identificación Personal (PIN)" [Online]. Disponible en: https://www.investopedia.com/terms/p/personal-identification-number.asp

BIOGRAFÍAS



Arroyo Christian, nació en la ciudad de Quito-Ecuador, se graduó en el Colegio Pedro Pablo Borja N°1, especialidad Físico – Matemático. En el año 2019 obtiene el título de Ingeniero en Electrónica y Redes de Información en la Escuela Politécnica Nacional. Actualmente, trabaja en el laboratorio electrónico Rosenkrantz y como profesor particular de programación en VB y C#. También trabajo para el Ministerio de Justicia, Derechos Humanos y Cultos como Asistente de Soporte TIC para el Proyecto de Dispositivos de Geo Posicionamiento.

Calderón Xavier nació en Quito-Ecuador, se graduó en el Colegio La Salle, especialidad Físico-Matemático. En el 1998 obtiene el título de Ingeniero en Electrónica y Telecomunicaciones en la Escuela Politécnica Nacional y en el 2002 se gradúa como Máster en Tecnologías de la Información en Fabricación en la Universidad Politécnica de Madrid. Actualmente, trabaja en la Escuela Politécnica Nacional y ha sido Miembro del Consejo de Departamento (Departamento de Electrónica y Redes de Información); es profesor principal a tiempo completo, Jefe del Laboratorio de Informática de la Facultad de Ingeniería Eléctrica y Electrónica y Director de Proyecto de Investigación Semilla.