

# Sistema multiplataforma de gestión de datos para visión por computadora

## Multiplatform data management system for computer vision

Irma-Delia Rojas-Cuevas<sup>1\*</sup> , Luis-Gerardo Méndez-López<sup>2</sup> , José-Rafael Mendoza-Vázquez<sup>3</sup> , Vicente Ramírez-Palacios<sup>3</sup> , Sergio-Javier Torres-Méndez<sup>4</sup> 

<sup>1</sup>Departamento de Sistemas y Computación

<sup>2</sup>División de estudios de Posgrado e investigación

<sup>3</sup>Departamento de Eléctrica y Electrónica

<sup>4</sup>Departamento de Metal Mecánica

<sup>1,2,3,4</sup>Instituto Tecnológico de Puebla

Av. Tecnológico 420, col. Maravillas. Puebla, Puebla, México. CP 72220

Correo-e: rojascid@yahoo.com.mx

PALABRAS CLAVE: RESUMEN

Sistema de gestión de datos para visión por computadora, sistema multiplataforma, ingeniería del software.

En este artículo se presenta el desarrollo de un sistema multiplataforma de gestión de datos para aplicación de visión por computadora. Se presenta la arquitectura del sistema, los componentes y los resultados del sistema en operación. El sistema tiene la finalidad de dar de alta, dar de baja, consultar y modificar los datos de los departamentos y del personal de una empresa, de igual forma registra la imagen del rostro del personal, esto, con el fin de realizar la identificación del personal para el sistema de asistencia. El sistema está integrado por un módulo de generación de base de datos un conjunto de tablas que contienen los datos de los trabajadores como son nombre, apellidos, departamento, entre otros; para el caso de los departamentos y trabajadores el sistema cuenta con un módulo de altas, un módulo de bajas, un módulo para modificación de datos; el sistema cuenta con un módulo para el registro de rostros de los trabajadores, un módulo para entrenamientos del algoritmo de reconocimiento facial y un módulo para identificación de rostro. Para el diseño del sistema se utilizó ingeniería del software. El sistema desarrollado tiene la capacidad de operar en Ubuntu y en Windows. Para el desarrollo del sistema se utilizó el lenguaje de programación Python y la librería OpenC.

KEYWORDS: ABSTRACT

Data management system for computer vision, multiplatform system, software engineering.

This article presents the development of a multiplatform data management system for computer vision application. The architecture of the system, the components and the results of the system in operation are presented. The system has the purpose of registering, canceling, consulting and modifying the data of the departments and workers of a company, in the same way it records the image of the face of the personnel, this, in order to carry out the identification of the workers for the assistance system. The system consists of a module for generation of database and a set of tables that contain the data of workers such as name, surname, department, among others; In the case of departments and workers, the system has a module for registration, a module for elimination, a module for data modification; The system has a module for register the face of workers, a module for training the facial recognition algorithm and a module for face identification. Software engineering was used for the design of the system. The developed system can be operated on Ubuntu and Windows. For the development of the system, the Python programming language and the OpenCV library were used.

Recibido: 4 de julio de 2021 • Aceptado: 23 de marzo de 2022 • Publicado en línea: 12 de agosto de 2022

# 1 INTRODUCCIÓN

Actualmente la seguridad en las empresas ha cobrado gran importancia, por lo que es necesario contar con sistemas que permitan la identificación de dicho personal al acceder o abandonar las instalaciones, de modo que puedan ser detectados los intentos de intrusión, en un formato cada vez más autónomo. Existe diversas propuestas para proveer seguridad de acceso a instalaciones, entre algunas de ellas podemos encontrar cerraduras biométricas (Acebo C. 2019) y reconocimiento facial (Cadena M. et al. 2017; Juanes 2019).

Este sistema fue desarrollado para dar una solución a la necesidad de las empresas de contar con sistema para el control de sus empleados, de tal forma que dicho control sea automatizado. Con este sistema se puede aprovechar la disponibilidad en las empresas de cámaras que permiten la captura de los usuarios. El sistema propuesto busca registrar en forma automatizada el ingreso del personal a un área por medio del reconocimiento facial del personal, una vez que se logra el registro del usuario se muestra un aviso, los datos del trabajador y su foto, cuando el trabajador ha sido identificado. Este sistema puede utilizarse con Windows o con Ubuntu, para que la empresa tenga la flexibilidad de elegir en que sistema operativo desea correr el Sistema de Gestión de Datos para Visión por Computadora.

El sistema fue programado con el editor de código Sublime Text, el lenguaje de programación utilizado fue Python y sus librerías Opencv y Tkinter. Se utilizaron Python y OpenCV debido a que son lenguajes amigables, con una gran potencialidad al usarlos y que permiten implementar códigos en corto tiempo (Khan et al. 2019; Wagner 2012).

Para la detección del rostro se usó el método HAAR cascade propuesto por (Viola and Jones 2001) y para el reconocimiento facial se usaron dos métodos, el método Local Binary Pattern Histogram (LBPH) propuesto por (Cheng et al. 2019) y el método Eigenfaces, desarrollado por Sirovich and Kirby (1987) y usado por primera vez por (Turk and Pentland 1991). Los métodos de reconocimiento facial Local Binary Pattern Histogram (LBPH) y Eigenfaces, son los métodos con mejor asertividad (Kortli et al. 2020; Kumar, Papendra, and Gupta 2019; Sharif et al. 2017).

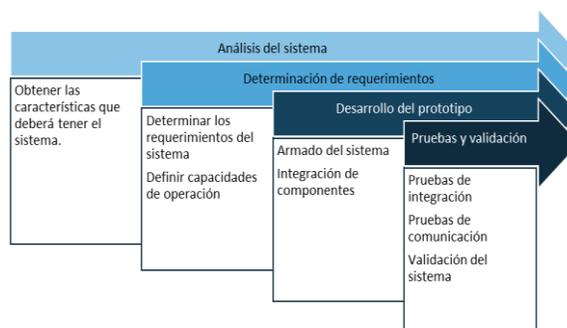
En lo que respecta a la base de datos. En este sistema, se usó una base de datos relacional que permite obtener los datos de diferentes tablas, estableciendo los

procedimientos de actualización y recuperación para facilitar la seguridad del conjunto de los datos (Moratalla, 2000). Para implementar las bases de datos se utilizó el software SQLite debido a sus características (Lacey 2019). SQLite posee ventajas mayores sobre otros administradores ya que tiene un peso ligero cuando se trata de una configuración compleja, gastos generales de administración, y el uso de recursos informáticos mínimos (Kreibich, 2010)

Para facilitar el uso del sistema se desarrolló una interfaz gráfica de usuario. En este punto se tomó en cuenta la coincidencia del diseño de la interacción y el de la interfaz de acuerdo con (Solano et al. 2011).

## 2 METODOLOGÍA

Para el desarrollo del sistema se utilizó una metodología de cuatro etapas, como puede observarse en la Figura 1. La primera etapa comprende el análisis del sistema, éste permite especificar el funcionamiento y las capacidades de operación. La segunda etapa es la determinación de requerimientos, en esta etapa se desarrollan los documentos de especificación de requerimientos de comunicación, interfaz, módulos a programar y pruebas de funcionamiento necesarias. La tercera etapa es el desarrollo del prototipo, donde se deberá armar el sistema. Finalmente, en la etapa cuatro se desarrollan las pruebas para verificar que el sistema no tiene problemas de comunicación y/o integración, que el sistema se comporta de forma adecuada y en caso necesario realizar ajustes.



El proceso de desarrollo se presenta en las siguientes secciones.

## 3 ANÁLISIS DEL SISTEMA

Como primera etapa se trabaja en el análisis del sistema. En esta etapa se determinarán los componentes necesarios para que el sistema sea funcional.

El sistema registra los datos del personal, también realiza el registro de acceso del personal, así como registrar los intentos de acceso del personal no registrado, para realizar estas funciones el sistema se compone por un módulo que genera una base de datos y las tablas para almacenar la información, también se cuenta con un módulo de almacenamiento para las imágenes que permite el entrenamiento de los algoritmos de reconocimiento facial y la fotografía del personal, así como almacenamiento de imágenes de las personas que intentaron acceder y no están registradas en el sistema. El sistema cuenta con módulos para dar de alta, baja, modificar los datos de los departamentos y del personal. Este esquema se muestra en la Figura 2.



Figura 2. Módulos necesarios en el sistema

#### 4 DETERMINACIÓN DE REQUERIMIENTOS

Como segunda etapa se tiene la determinación de requerimientos del sistema. En esta etapa se detallarán los componentes que se requieren para el sistema. Como requerimientos, se tiene: el sistema debe funcionar con sistema operativo Windows o con Ubuntu. El sistema interactúa con bases de datos para el registro de los datos de los trabajadores y el registro de asistencia, así como el registro de intentos de acceso no autorizado. El sistema cuenta con la opción de agregar, modificar, consultar, dar de baja departamentos y trabajadores. El sistema trabaja con un almacenamiento interno para la identificación de los trabajadores. El sistema trabaja con dos perfiles de usuario, un administrador y un supervisor, el supervisor interactúa con el registro manual de acceso y a la identificación de los trabajadores, el administrador accede a la identificación de los trabajadores y además

genera la base de datos y las tablas, dar de alta, dar de baja o realizar la modificación de los datos del personal y de los departamentos. Estos requerimientos se presentan en el diagrama de casos de uso de la Figura 3 y permiten ver como interactúan los actores con los diferentes casos de uso.

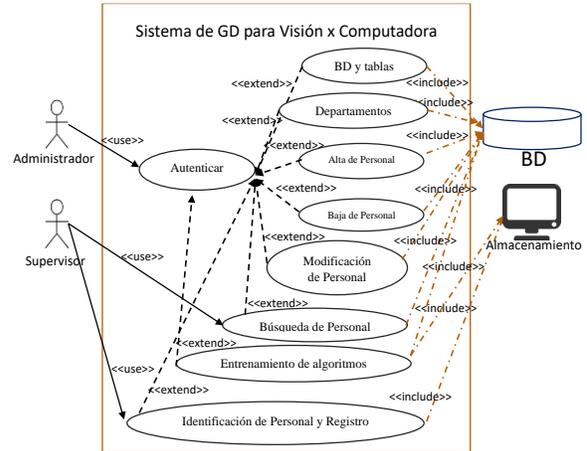


Figura 3. Diagrama de casos de uso

### 5 DESARROLLO DEL PROTOTIPO

En la etapa de desarrollo del prototipo, se presentan los lenguajes utilizados para su programación y los sistemas operativos para los que se programó el sistema, el modelado de la base de datos, el desarrollo de las interfaces gráficas y la adquisición de imágenes e identificación.

#### 5.1 LENGUAJES DE PROGRAMACIÓN Y SISTEMAS OPERATIVOS UTILIZADOS

El sistema de gestión de la información se programó utilizando el editor Sublime Text en su versión 3.2.2 y utilizó el lenguaje Python en su versión 3.9.2, también la librería OpenCV en su versión 3.8.5. Estos elementos permiten construir el sistema debido a la capacidad y versatilidad que tienen al interactuar de forma ágil con los sistemas operativos Windows y Ubuntu. Además de lo citado, se emplearon las librerías libres hechas para Python: Tkinter, Pillow, Os, Numpy, Imutils, Pyexe, sqlite3, que entre otras acciones permiten desarrollar interfaces gráficas, cálculos, análisis matemáticos, interactuar con subdirectorios y bases de datos. Para la verificación de la base de datos y su funcionamiento se utilizó el software SQLite en su versión 3.32.2. Los sistemas operativos en los que se probó el sistema fueron Windows 10 Pro y Ubuntu 20.04. En ambos se verificó el funcionamiento del sistema con resultados correctos y

operación sin problemas. Como siguiente etapa se desarrolló el Modelado de la base de datos en sqlite, la cual se detalla a continuación.

## 5.2 MODELADO DE LA BASE DE DATOS

El modelado de la base de datos se muestra en la Figura 4, se tomó como base, los requerimientos definidos en la sección 4 y como resultado se obtuvieron las entidades con sus atributos y la relación entre entidades. Para el caso de la base de datos, un departamento puede tener varios empleados y un empleado puede tener varios accesos. Los registros de entrada se registran en la tabla ACCESO cuando el empleado ingresa a la empresa, pero si la persona que intenta ingresar a la empresa no se encuentra registrada, se guarda el intento de acceso en la tabla NOACCESO.

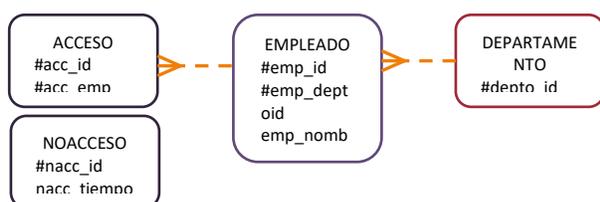


Figura 4. Modelado de la base de datos

La base de datos cuenta con cuatro tablas. La tabla departamento (Ver Tabla 1) tiene como identificador único a una llave primaria que es el número de departamento, también cuenta con el nombre del departamento. La tabla empleado (Ver Tabla 2) tiene un identificador único o una llave primaria que es el número de empleado, una llave foránea que relaciona al trabajador con el departamento, en la tabla también pueden registrarse en nombre del trabajador, sus apellidos, su dirección y su cargo. La tabla de acceso (Ver Tabla 3) tiene una llave primaria de tipo entero y autoincremental, una llave foránea que relaciona el acceso con el trabajador autoincremental y un campo que registra la fecha y hora de asistencia. La tabla no acceso (Ver Tabla 4) tiene una llave primaria entera autoincremental y un campo que registra la fecha y hora del intento de acceso.

Campo	Tipo	Tamaño	Descripción
depto_id	int	4	Identificador único por departamento
depto_nombre	varchar	30	Nombre del departamento

Tabla 1. Tabla departamento

Campo	Tipo	Tamaño	Descripción
emp_id	int	4	Identificador único de empleado que se relaciona con la tabla acceso
emp_deptoid	Int	4	Departamento donde labora el empleado que se relaciona con la tabla departamento
emp_nombre	varchar	30	Nombre del empleado
emp_apellidos	varchar	30	Apellidos del empleado
emp_direccion	varchar	30	Dirección del empleado
emp_cargo	varchar	30	Puesto del empleado
emp_foto	varchar	70	Dirección dónde se almacenó la foto del empleado y nombre del archivo

Tabla 2. Tabla empleada

Campo	Tipo	Tamaño	Descripción
acc_id	int	4	Identificador único en cada registro de acceso
acc_emp	int	4	Permite relacionar registro de acceso con el empleado
acc_tiempo	timestamp		Fecha y hora de acceso

Tabla 3. Tabla acceso

Campo	Tipo	Tamaño	Descripción
nacc_id	int	4	Identificador único de los registros de acceso no autorizado
acc_tiempo	timestamp		Fecha y hora de accesos no autorizados

Tabla 4. Tabla no acceso

## 5.3 DESARROLLO DE LAS INTERFACES GRÁFICAS

La interfaz gráfica del sistema se diseñó en forma de “notebook” comúnmente llamado cascada, en donde se tienen 6 pestañas y estas son BD, Departamentos, Trabajadores, Registro de rostro, Entrenamiento e Identificación. En la pestaña BD se inicializa la base de datos con las tablas. En la pestaña Departamentos se tiene

acceso a la gestión de los departamentos, para dar de alta, dar de baja, modificar y mostrar departamentos. En la pestaña trabajadores se tiene acceso a la gestión de los trabajadores, para dar de alta, dar de baja, modificar, buscar un trabajador, mostrar todos los trabajadores registrados. En la pestaña registro de rostro, se capturan fotografías tomadas por la cámara y se almacenan en el disco duro. En la pestaña Entrenamiento se entrena a los algoritmos con las imágenes de todos los trabajadores y se obtiene un modelo para cada algoritmo y se guardan en el disco duro. En la pestaña Identificación hace el reconocimiento del rostro del trabajador para que se realice el registro de acceso a la empresa, si el trabajador es reconocido se muestran sus datos y su fotografía en la pantalla, de lo contrario se muestra el mensaje de que la persona no ha sido identificada, se toma una fotografía del "intruso" que se almacena en el disco duro y se hace el registro en la base de datos del intento de acceso con su estampa de tiempo.

### 5.4 ADQUISICIÓN DE IMÁGENES E IDENTIFICACIÓN

Para cada trabajador y lograr su identificación por reconocimiento facial se capturaron *frames* o cuadros a razón de 24 fps, durante 8.2 segundos, obteniéndose un total de 199 imágenes, las cuales permiten entrenar al sistema. Para el entrenamiento de los algoritmos se utilizan las imágenes guardadas de cada trabajador, y se generan dos modelos, uno por cada método de reconocimiento empleado. Los modelos se generan en formato XML los cuales tiene los datos numéricos del reconocimiento facial del trabajador. En la Figura 5 se muestra el ejemplo de los modelos LBPH Y eigenfaces

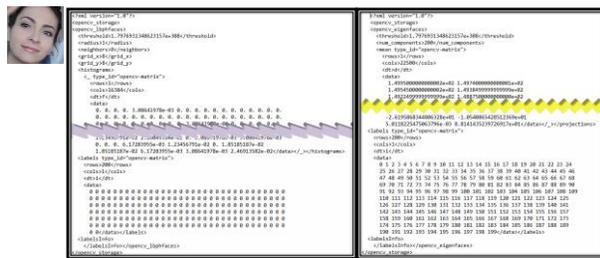


Figura 5. Ejemplo de modelos LBPH y eigenfaces.

Para la identificación del personal, se verificó en tiempo real que el programa identifique los rostros del personal, tomando como base los modelos creados en XML, y en forma posterior se genera el registro en la tabla ACCESO con el número de empleado, la fecha y hora. Cuando el

personal no es identificado, se generaba un registro con la fecha y hora en la tabla NOACCESO.

## 6 PRUEBAS Y RESULTADOS DEL SISTEMA

Con los componentes anteriores se verificó la operación del sistema en los sistemas operativos Windows y Ubuntu. Lo primero que se verificó fue que la cámara captara las imágenes de los trabajadores y las enviara a la computadora al momento de hacer el registro de rostro. También se verificó que el sistema pudiera hacer el registro, modificación, baja y consulta de personal y departamentos. Se verificó que se generara el modelo para cada método utilizado. Finalmente se verificó que el sistema pudiera realizar la identificación del personal y el registro de acceso, así como los registros de acceso no autorizados. En la Figura 6 se muestra la computadora con Windows y la computadora con Ubuntu, ambas con el sistema corriendo.



Figura 6. Sistema de registro de personal en Windows y Ubuntu.

En la Figura 7 se muestran la interfaz de gestión de trabajadores, dónde se cuenta con las opciones de dar de alta, buscar, modificar, borrar y la opción de mostrar todos los trabajadores registrados en la *base de datos*.



Figura 7. Interfaces gráficas del Sistema de gestión de datos

Se realizaron pruebas de asertividad al sistema para el método LBPH y el método Eigenfaces en dos etapas diferentes.

### ETAPA 1. INTRUSOS Y PERSONAL

En las pruebas se obtuvieron identificaciones que se clasificaron de acuerdo con los siguientes criterios:

- Verdaderos positivos. Es la cantidad de trabajadores que estaban registrados y fueron correctamente identificados.
- Falsos negativos. Es la cantidad de trabajadores que estaban registrados, pero no fueron identificados.
- Falsos positivos. Es la cantidad de personas no registradas (intrusos) que fueron identificados como trabajadores.
- Verdaderos negativos. Es la cantidad de personas no registradas (intrusos) que no fueron identificados.

Como resultado del método LBPH, este arrojó que de 100 personas registradas como trabajadores fueron reconocidas 98 y 2 no lo fueron. Con este mismo método los intrusos, es decir, personas no registradas cuyo total fueron 20, arrojó un resultado de 18 personas correctamente identificadas como no reconocidas y 2 personas fueron confundidas con trabajadores registrados. Ahora bien, para el algoritmo Eigenfaces se obtuvo un resultado de 92 personas reconocidas correctamente y 8 personas no reconocidas, mientras que, de 20 personas no registradas, se tuvieron 16 personas no identificadas y 4 personas confundidas con trabajadores, como se muestra en la Figura 8.

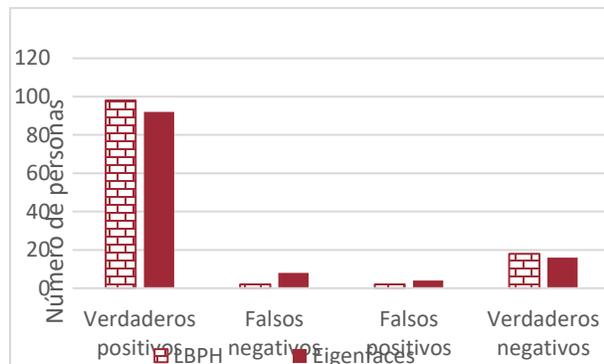


Figura 8. Reconocimiento facial de trabajadores e intrusos

## ETAPA 2. RASGOS SIMILARES POR FAMILIARIDAD

Para el caso en el que haya dos hermanos cuyos rasgos son similares, se utilizaron las imágenes de 24 personas correspondientes a 12 parejas de hermanos.

En las pruebas se obtuvieron identificaciones que se clasificaron de acuerdo con los siguientes criterios:

- Verdaderos positivos. Es la cantidad de personas que estaban registradas y fueron identificadas correctamente.
- Falsos negativos. Es la cantidad de personas que estaban registradas pero que no fueron reconocidas.
- Falsos positivos. Es la cantidad de personas que fueron confundidas con otra persona.
- Verdadero negativo. Es la cantidad de personas no registradas y no identificadas.

El método LBPH reconoció correctamente a catorce personas, no identificó a dos personas y a ocho de ellas las confundió con su hermano. Mientras que el método Eigenfaces reconoció correctamente a diez personas, a dos de ellas no las reconoció y a doce de ellas las confundió con su hermano. Cabe resaltar que las imágenes utilizadas son de gemelos. Los resultados se muestran en la Figura 9.

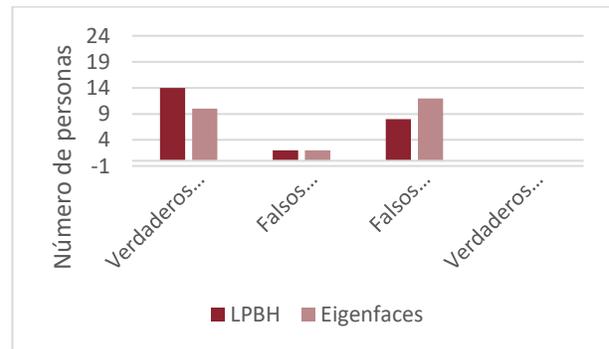


Figura 9. Reconocimiento facial de rasgos similares por familiaridad

## 7 CONCLUSIONES Y TRABAJO FUTURO

Se desarrolló un sistema de gestión de datos para aplicaciones de visión por computadora con capacidad de operar en Windows y en Ubuntu. El sistema fue desarrollado en Python con OpenCV debido a que ambos pueden trabajar en los dos sistemas operativos mencionados de forma ágil. El sistema se desarrolló con base a los requerimientos establecidos en una empresa que desea administrar el ingreso del personal y mediante la identificación con imágenes y su registro. Este sistema tiene el propósito de permitir el registro de acceso de personal y de identificar los intentos de acceso no autorizados. El sistema fue probado en Windows y Ubuntu, el desempeño en ambos sistemas operativos fue correcto.

El sistema administra los datos de los trabajadores y de los departamentos de una empresa, de igual forma

registra el rostro de los trabajadores para identificarlos y registrar la asistencia. El sistema de gestión de datos contempla el uso de una base de datos para guardar la información del personal, guardar los intentos de acceso no autorizado y el registro de asistencia. El sistema contempla diferentes interfaces de usuario en formato de ventanas para el uso de los diferentes usuarios del sistema, la navegación y su uso.

De los métodos utilizados para el reconocimiento facial, el método de LBPH obtuvo una asertividad del 97%, que contempla los resultados de verdaderos positivos y verdaderos negativos de la Etapa 1 dónde se identifica personal y a los intrusos; mientras que con el método Eigenfaces, en la misma etapa se obtuvo una asertividad del 90%. En la etapa 2, dónde se identifican personas con rasgos familiares, el método LBPH obtuvo una asertividad del 91%, mientras que el método Eigenfaces obtuvo una asertividad del 92%. Por lo que el método LBPH es el que tiene mejor asertividad al momento de identificar a los trabajadores, mientras que el método Eigenfaces tiene mejor asertividad al momento de identificar personas con rasgos familiares.

Como trabajo futuro se va a desarrollar la identificación de vista frontal, lateral izquierda y lateral derecha combinadas para la identificación de las personas, a fin de mejorar el asertividad de los métodos.

## REFERENCIAS

- Acebo C. M. M. Implementación de acceso inteligente a las oficinas de coordinación y sala de docentes en la carrera de ingenierías en sistemas computacionales. Recuperado el 01 de febrero de 2021, de <http://repositorio.unesum.edu.ec/handle/53000/1522>, 2019.
- Cadena, M. J. A. Montaluisa, P. R. H., Flores L. G. A., Chancúsig C. J. C., Guaypatín P. O. A. Reconocimiento Facial Con Base En Imágenes. *Boletín Virtual Redipe*. 2017. 6(5), 143-151.
- Juanes, Gemma G. El Aeropuerto de Menorca, Pionero En Sistema de Reconocimiento Facial. Cuadernos de Seguridad. Recuperado el 27 de febrero de 2021, de <https://cuadernosdeseguridad.com/2019/04/e1-aeropuerto-de-menorca-pionero-en-sistema-de-reconocimiento-facial/>. 2019.
- Khan, M., Sudeshna, C., Rani A., Shaveta K. Face Detection and Recognition Using OpenCV. *International Conference on Computing, Communication, and Intelligent Systems, (ICCCIS)*. 2019, 116-119. Disponible en: doi: 10.1109/ICCCIS48478.2019.8974493.
- Wagner, P. Face Recognition with Python. Tersedia Dalam: Www. Bytefish de [Diakses Pada]. Recuperado el 10 de diciembre de 2020, [https://www.bytefish.de/pdf/facerec\\_python.pdf](https://www.bytefish.de/pdf/facerec_python.pdf), 2012.
- Kortli, Y., Maher, J., Ayman, A. F., Mohamed, A. Face Recognition Systems: A Survey. *Sensors (Basel, Switzerland)*. 2020. Disponible en: doi: 10.3390/s20020342
- Kumar, R., Papendra, K., Abhishek G. Review Paper on Face Recognition Techniques. *International Journal of Computer Sciences and Engineering*. 2019, 7(5), 223–29..
- Sharif, M., Naz, F., Yasmin, M., Shahid, M. A., Rehman, A. Face Recognition: A Survey. *Journal of Engineering Science and Technology Review*. 2017, 10(2), 166-177.
- Cheng, E. J., Chou, K. P., Rajora, S., Jin B. J., Tanveer, M., Lin, C. T., Young, K. Y., Lin, W. C., Prasad, M. Deep Sparse Representation Classifier for Facial Recognition and Detection System. *Pattern Recognition Letters*. 2019, 125, 71–77. Disponible en: doi: 10.1016/j.patrec.2019.03.006.
- Viola, P., Jones, M. Rapid Object Detection Using a Boosted Cascade of Simple Features. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2001, 1, 1-9. Disponible en: doi: 10.1109/CVPR.2001.990517
- Turk, M. A., Pentland, A. P. Face recognition using eigenfaces. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*. 1991, 586-587.
- Moratalla, J. Base de datos con SQL server 2000. Recuperado el día 15 de diciembre de 2020. [https://issuu.com/ryuuga/docs/bases\\_de\\_datos\\_con\\_sql.server\\_2000\\_-\\_transact\\_sql/193](https://issuu.com/ryuuga/docs/bases_de_datos_con_sql.server_2000_-_transact_sql/193), 2001
- Lacey, N. Python by Example. Recuperado el día 10 de diciembre de 2020, <https://www.cambridge.org/core/books/python-by-example/sqlite/6213919377195F9D0510515F328C6EE1>, 2019. Disponible en: doi:10.1017/9781108591942.021
- Kreibich, J. Using SQLite. Sebastopol, California: O'Reilly Media, Inc., 2010
- Solano, A., Chanchí, G. E., Collazos, C. A., Arciniegas, J. L., Rusu, C. Diseñando Interfaces Gráficas Usables de Aplicaciones En Entornos de Television Digital Interactiva. *Asociation for Computer Machinering*. 2011, 7811, 366-375.

*Acerca de los autores*

**Irma Delia Rojas Cuevas**, recibió el grado de Doctora en Logística y Dirección de la Cadena de Suministro con mención honorífica en 2019 por parte de la Universidad Popular Autónoma del Estado de Puebla y el grado de Maestra en Ciencias en Ingeniería Industrial por parte del Instituto Politécnico Nacional, sección UPIICSA, en el año 2000, con la especialización en Manufactura. Actualmente, se desempeña como profesor de tiempo completo en el Tecnológico Nacional de México/ Instituto Tecnológico de Puebla para el Departamento de Sistemas y Computación. Es miembro del Cuerpo Académico Sistemas interactivos y realidad virtual. Su investigación se encuentra en el área de sistemas interactivos y realidad virtual con énfasis en sistemas mecatrónicos.



**Luis Gerardo Méndez López**, recibió el grado de Ingeniero en Electrónica con la especialización en automatización de procesos industriales en el año 2018 por el Tecnológico Nacional de México/ Instituto Tecnológico de Puebla, actualmente se encuentra estudiando la maestría en ingeniería electrónica en la LGAC sistemas mecatrónicos interactivos aplicados al control de procesos en el Tecnológico Nacional de México/ Instituto Tecnológico de Puebla. Ha trabajado en proyectos de telecomunicaciones para la empresa Huawei y Altan. También se ha desempeñado como consultor de proyectos en la misma área y ha trabajado en el área de mantenimiento industrial como técnico de mantenimiento nivel A.



**José Rafael Mendoza Vázquez**, recibió el grado de Doctor en Ciencias en Electrónica en 2010 por parte del Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) y el grado de Maestro en Ciencias en Electrónica en 2003 por la misma institución. Es profesor de tiempo completo en el Tecnológico Nacional de México/ Instituto Tecnológico de Puebla, en el departamento de Ingeniería eléctrica y electrónica. Es miembro del Cuerpo Académico Sistemas Interactivos y Realidad Virtual. Su investigación se encuentra dentro el área de robótica y control con énfasis en robótica, control de

movimiento, modelado, procesamiento de señales y sistemas interactivos.



**Sergio Javier Torres Méndez**, recibió el grado de Doctor en Ciencias en Ingeniería Mecánica en 2014 por parte de la Universidad de Waterloo, Ontario, Canadá; el grado de Maestro en Ciencias en Electrónica en 2008 por la Benemérita Universidad Autónoma de Puebla, y el grado de Maestro en Ciencias en Ingeniería Mecánica por parte del Instituto Tecnológico de Veracruz en 1995. Es profesor de tiempo completo en el Tecnológico Nacional de México/ Instituto Tecnológico de Puebla desde 1995. Es miembro del Cuerpo Académico Sistemas Interactivos y Realidad Virtual. Sus intereses de investigación se relacionan con el diseño y desarrollo de robots de alta velocidad y de sistemas mecatrónicos interactivos para la asistencia humana.



**Vicente Ramírez Palacios**, recibe el grado de Maestro en Ciencias en Electrónica en 1997 por el Instituto Nacional de Astrofísica, Óptica y Electrónica y el título de Ingeniero Industrial en Electrónica en 1996 por el Tecnológico Nacional de México/ Instituto Tecnológico de Puebla. Es profesor de tiempo completo a nivel Licenciatura en el Instituto Tecnológico de Puebla. Ha trabajado en Automatización y Control de Procesos, utilizando algoritmos PID, Lógica Difusa y Redes Neuronales; comunicando los procesos a través de Redes Industriales tales como: Red ASi, Red Profibus, Red Ethernet y Red Profinet, teniendo Control y Adquisición de Datos de los procesos a través de un SCADA. Es integrante del cuerpo académico Sistemas Interactivos y Realidad virtual.