



**UNIVERSIDAD TECNOLÓGICA ISRAEL**

**TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE:**

**INGENIERO EN ELECTRÓNICA DIGITAL Y TELECOMUNICACIONES**

**TEMA: SISTEMA DE VIDEO VIGILANCIA MEDIANTE VISIÓN POR  
COMPUTADOR PARA EL CENTRO DE EDUCACIÓN INICIAL N°1 DEL  
MINISTERIO DE EDUCACIÓN.**

**AUTOR: Leonardo Rubén Arellano Arellano**

**TUTOR: Ing. René Ernesto Cortijo Leyva, Msc**

**QUITO- ECUADOR**

**AÑO: 2020**

**UNIVERSIDAD TECNOLÓGICA ISRAEL**

**APROBACIÓN DEL TUTOR**

En mi calidad de tutor del trabajo de titulación certifico:

Que el trabajo de titulación “**SISTEMA DE VIDEO VIGILANCIA MEDIANTE VISIÓN POR COMPUTADOR PARA EL CENTRO DE EDUCACIÓN INICIAL N°1 DEL MINISTERIO DE EDUCACIÓN**”, presentado por el **Sr. Leonardo Rubén Arellano Arellano**, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito D.M. 02 marzo de 2020

TUTOR

.....

Ing. René Ernesto Cortijo, Msc

## **AGRADECIMIENTO**

Agradezco a toda mi familia por su apoyo y comprensión, al Ing. René Ernesto Cortijo Leyva, por su invaluable dedicación y guía en este proyecto, a los Ingenieros Francisco Jurado, Fidel Parra, David Cando, por las valiosas sugerencias en la elaboración del documento final, a la Universidad Tecnológica Israel, al cuerpo docente que lo conformaron y que aún lo conforman, a los Ing. Carlos Pardo, Eduardo Pasochoa, Gabriel Alvear; grandes personas que contribuyeron con su experiencia y conocimientos al desarrollo y culminación de este trabajo, al Ministerio de Educación, Ing. Esteban Aguirre, Magister Evelyn Espinosa y todo el valioso cuerpo docente de tan prestigiosa comunidad educativa donde se desarrolló el proyecto, de todo corazón, un especial y noble agradecimiento en estas humildes palabras.

Finalmente, a Dios, la ciencia y la madre naturaleza por una oportunidad más en la vida para explorar sus mas infinitos secretos y plasmarlos en papel para las futuras generaciones por que la visión artificial no es una de las cosas más fáciles de hacer, pero cada vez estamos más cerca de conseguirlo.

## **DEDICATORIA**

Sin duda alguna, a mis amados y queridos padres, José y Francisca Arellano, que siempre estuvieron en las buenas y en las malas dando y guiándome con su apoyo y consejos incondicionales, a mi hermosa familia, Sofía y Kaory, por entenderme, apoyarme y ser esa luz de felicidad a mil años Luz, a ti hermanita querida por ser mi ejemplo de progreso a seguir, como decimos en nuestro pueblo ancestral:

**Diusulpagui, Tukuy shunguan, cangunata kay uchina gestuta alichipaychi cangunapa shungupi.**

Dios les pague con todo el cariño y corazón les dedico para que lo lleven en su corazón.

## ÍNDICE

CAPÍTULO 1 .....	7
FUNDAMENTACIÓN TEÓRICA .....	7
1.1. SISTEMAS DE VIDEO VIGILANCIA .....	7
1.1.1. Evolución de los sistemas de video vigilancia.....	7
1.1.2. Tipos de sistemas de video vigilancia.....	8
1.1.3. Cámaras de video.....	9
1.1.4. Características técnicas de una cámara de video vigilancia.....	12
1.1.5. Estándar de compresión .....	14
1.1.6. Sistemas de control de acceso.....	15
1.2. VISIÓN POR COMPUTADOR.....	20
1.3.1. Procesamiento de imágenes .....	21
1.3.2. Etapas del procesamiento de imágenes.....	23
1.3. INTELIGENCIA ARTIFICIAL.....	27
1.3.3. Machine Learning .....	29
1.4. DEEP LEARNING .....	32
1.4.1. Redes neuronales .....	32
1.4.2. Redes neuronales multicapa.....	35
1.4.3. Redes neuronales convolucionales .....	37
1.5. RECONOCIMIENTO DE ROSTROS.....	40
1.5.1. Reconocimiento de rostros a partir del algoritmo de Viola-Jones.....	41
1.5.2. Reconocimiento facial a partir de histograma de gradientes HOG.....	47

1.6. DETECCIÓN DE OBJETOS.....	58
1.7. LEGISLACIÓN. ....	60
CAPÍTULO II.....	63
MARCO METODOLÓGICO .....	63
2.1. METODOLOGÍA DE INVESTIGACIÓN, PROCEDIMIENTOS Y HERRAMIENTAS.....	63
2.1.1. Metodología de la investigación .....	63
2.1.2. Procedimientos.....	64
2.1.3. Herramientas.....	65
CAPÍTULO III .....	66
PROPUESTA .....	66
3.1. RECONOCIMIENTO FACIAL .....	67
3.2. HERRAMIENTAS DE SOFTWARE.....	74
3.2.1. Lenguaje de programación Python .....	74
3.3. ANÁLISIS DE COSTOS Y TIEMPO .....	79
3.4. VENTAJAS DEL PRODUCTO .....	80
CAPÍTULO IV .....	92
IMPLEMENTACIÓN .....	92
4.1. DESARROLLO .....	92
4.1.1. Desarrollo del software de reconocimiento de rostros.....	92
4.1.2. Circuito de control de la puerta de acceso. ....	93
4.1.3. Desarrollo del software de control de la tarjeta arduino.....	94
4.1.4. Diseño, construcción y ensamble de cajas.....	94

4.1.5. Desarrollo del software de reconocimiento de personas. ....	95
4.1.6. Instalación de cámaras, dispositivos y red. ....	95
4.2. SEÑALIZACIÓN.....	96
4.2.1. Puesta en marcha del ingreso de Usuarios a la base de datos del sistema. .	97
4.2.2. Puesta en marcha del sistema de respaldo energético para el control de la puerta de acceso .....	98
4.2.3. Puesta en marcha del sistema de detección de personas.....	98
4.3. PRESUPUESTO .....	99
4.4. PRUEBAS DE FUNCIONAMIENTO .....	102
4.3.1. Pruebas de funcionamiento del sistema de control de acceso.....	102
4.3.2. Pruebas de funcionamiento del sistema de reconocimiento de personas..	102
4.3.3. Pruebas de funcionamiento del sistema de detección de personas .....	103
4.3.4. Interfaz de funcionamiento del sistema cctv mediante visión por computadora.....	104
4.5. ANÁLISIS DE RESULTADOS .....	105
CONCLUSIONES.....	109
RECOMENDACIONES .....	111
BIBLIOGRAFÍA.....	112

## ÍNDICE DE FIGURAS

Figura. 1.1. Cámara CCTV tipo mini bala para interiores .....	9
Figura. 1.2. Cámara para exterior IP66 .....	9
Figura. 1.3. Cámara antivandálica fija para exterior .....	10
Figura. 1.4. Cámara CCTV IP .....	10
Figura. 1.5. Cámara WiFi .....	11
Figura. 1.6. Cámara 10X zoom óptico .....	11
Figura. 1.7. Cámara oculta portátil WiFi .....	12
Figura. 1.8. Cámara IP, 3G, 4G, WiFi, tarjeta SD, audio .....	12
Figura. 1.9. Rango de unidades de iluminación en Lux .....	13
Figura. 1.10. Tamaño de la imagen en función de MP .....	13
Figura. 1. 11. Comparativa entre a) Cámara analógica de 600TVL y .....	14
Figura. 1.12. Comparación tasa de compresión .....	15
Figura. 1.13. Sensor óptico de huellas digitales .....	17
Figura. 1. 14. Sensor óptico de huellas digitales .....	18
Figura. 1.15. Técnicas de reconocimiento facial .....	19
Figura. 1. 16. Etapas del procesamiento de imágenes .....	24
Figura. 1.17. Tipos de imagen para procesamiento .....	24
Figura. 1.18. Eliminación de ruido .....	25
Figura. 1.19. División de imágenes en regiones .....	25
Figura. 1.20. Obtención de contornos y rasgos característicos .....	26
Figura. 1.21. Reconocimiento digital de rostro mediante algoritmos de aprendizaje .....	27
Figura. 1.22. Aplicaciones de la inteligencia artificial .....	28
Figura. 1. 23. Algoritmo de aprendizaje supervisado .....	29
Figura. 1. 24. Algoritmo de aprendizaje no supervisado .....	30
Figura. 1.25. Aprendizaje de un juego mediante algoritmo ML por refuerzo .....	30
Figura. 1.26. Clasificación de los algoritmos de Machine Learning .....	31
Figura. 1. 27. Ejemplo de una red neuronal .....	33
Figura. 1. 28. Comparación entre una neurona biológica y una artificial .....	33
Figura. 1. 29. Ejemplo de una red neuronal .....	35



Figura. 1.30. Proceso de convolución bit a bit.....	38
Figura. 1.31. Proceso de pooling en una matriz 4x4, resultante 2x2 .....	39
Figura. 1. 32. Capas convolucionales y de pooling en una CNN.....	40
Figura. 1.33. Reconocimiento de rostros a partir del algoritmo de Viola-Jones.....	41
Figura. 1.34. Reconocimiento de los ojos mediante AdaBoost .....	42
Figura. 1. 35. Imágenes de entrenamiento .....	44
Figura. 1.36 Imágenes Eigenfaces .....	45
Figura. 1.37. Imágenes Eigenfaces y Fisherfaces .....	46
Figura. 1.38. Análisis de rostros mediante histograma de gradientes orientados .....	47
Figura. 1.39. Conversión a) Original b) Escala de grises.....	48
Figura. 1.40. a) Imagen original, b) Rotación de la imagen.....	48
Figura. 1.41. Diagrama de gradientes orientados.....	49
Figura. 1. 42. Procesamiento HOG de la imagen.....	49
Figura. 1. 43. Diagrama de gradientes aplicado a un rostro.....	50
Figura. 1.44. Comparación de gradientes.....	51
Figura. 1. 45. Puntos de referencia facial.....	52
Figura. 1.46. Puntos de referencia facial en imagen de prueba.....	52
Figura. 1.47. Uso de puntos de referencia facial para normalizar un rostro .....	53
Figura. 1. 48. Entrenamiento de la red neuronal convolucional .....	54
Figura. 1.49. Entrenamiento de la red neuronal convolucional .....	55
Figura. 1. 50. Base de datos para entrenamiento ResNet-34 .....	55
Figura. 1. 51. Clasificador SVM.....	56
Figura. 1.52. Ejemplo del clasificador K-Nearest Neighbours KNN .....	57
Figura. 1.53. Reconocimiento de personas y objetos con tensorflow .....	60
Figura. 3.1. Diagrama de bloques general del sistema de cctv .....	69
Figura. 3.2. Flujograma de entrenamiento de conjunto de datos .....	70
Figura. 3.3. Flujograma de reconocimiento de rostros.....	70
Figura. 3.4. Flujograma de control mediante arduino .....	71
Figura. 3.5. Flujograma de reconocimiento de personas .....	72
Figura. 3.6. Cálculo de espacio en disco duro para grabación de imágenes y video .....	74

Figura. 3.7. Instalación de Python y Anaconda .....	75
Figura. 3.8. Creación del entorno virtual face recognition .....	77
Figura. 3.9. IDE Spyder .....	78
Figura 4.1 Numero mínimo de fotografías para entrenamiento .....	93
Figura 4.2 Control eléctrico de cerradura de la puerta de acceso .....	93
Figura 4.3 Circuito electrónico de control de la puerta de acceso .....	94
Figura 4.4 Módulo electrónico de retroalimentación, posicionamiento de usuario .....	95
Figura 4.5 Letrero de Información .....	97
Figura 4.6 Ingreso de nuevos usuarios.....	97
Figura 4.7 Respaldo Energético .....	98
Figura 4.8 Proceso de detección de personas.....	98
Figura 4.9 Módulo de retroalimentación para ubicación de rostro e ingreso.....	102
Figura 4.10 Sistema reconociendo a personas no autorizadas. ....	103
Figura 4.11 Detección de personas .....	103
Figura 4.12 Interfaz para visualización del guardia de turno.....	104
Figura 4.13 Personal registrado utilizando accesorios para la cabeza .....	105
Figura 4.14 Personal identificado y almacenado en la base de datos con lentes. ....	106
Figura 4.15 Proceso de identificación y detección de personas parte 1 .....	106
Figura 4.16 Proceso de identificación y detección de personas pate 2. ....	107
Figura 4.17 Teclado para ingreso de clave para apertura de puerta.....	107
Figura 4.18 Validación del sistema en horario diurno. ....	108
Figura 4.19 Detección de personas horario nocturno.....	108

## ÍNDICE DE TABLAS

Tabla. 1.1. Funciones de activación neuronal .....	34
Tabla. 1.2. Características Haar .....	433
Tabla. 1.3. Resultados de reconocimiento .....	46
Tabla. 1.4. Presupuesto .....	99

## RESUMEN

Esta investigación se basa en la implementación de un sistema de video vigilancia mediante visión por computador con aplicaciones determinadas dentro del área de reconocimiento facial y detección de personas, que utiliza una base de datos de imágenes para el preentrenamiento del sistema mediante técnicas de Machine Learning y Deep Learning. Se presenta como un proyecto factible que propone una solución a la falta de un circuito de CCTV para contrarrestar problemas debido al bajo nivel de seguridad sobre todo en el ingreso principal al establecimiento ante personas no autorizadas a la entidad educativa. El objetivo del presente proyecto es implementar un sistema de videovigilancia confiable que permita reducir el riesgo de seguridad existente. Se optó por utilizar nuevas tendencias tecnológicas enfocadas al área de inteligencia artificial, las cuales implican una optimización del uso de recursos, universalidad y mayor confiabilidad. Para este caso se optó por usar la visión por computador para estructurar un CCTV. El software que se elaboró para el circuito CCTV con reconocimiento facial y detección de personas emplea un conjunto de cámaras digitales, controladas por tres pcs por el nivel de procesamiento que este requiere, el cual está acoplado a un sistema electrónico mixto, que controla una cerradura magnética. El software de reconocimiento facial y el software de detección de personas están vinculados a un pc que hace el papel de servidor, en el que ejecuta un desarrollo en Python para que se muestre al guardia de turno, solo las imágenes de las cámaras, sin que este tenga acceso a la manipulación o modificación de las cámaras o del sistema en general.

**PALABRAS CLAVES:** Apertura automática, tensorflow, Python, reconocimiento facial, video vigilancia, visión por computador.

## **ABSTRACT**

This research is based on the implementation of a video surveillance system using computer vision with specific applications in the area of facial recognition and detection of people, which uses an image database for the pretraining of the system using machine learning and deep learning techniques. It is presented as a feasible project that proposes a solution to the lack of a CCTV circuit to counteract problems with the low level of security, especially at the main entrance to the establishment from persons not authorized to the educational entity. The objective of this project is to implement a reliable video surveillance system that will reduce the existing security risk. The decision was made to use new technological trends focused on the area of artificial intelligence, which imply an optimization of the use of resources, universality and greater reliability. In this case, the decision was made to use computer vision to structure a CCTV. The software that was developed for the CCTV circuit with facial recognition and detection of people uses a set of digital cameras, controlled by three PCs for the level of processing required, which is coupled to a mixed electronic system, which controls a magnetic lock. The facial recognition software and the people detection software are linked to a pc that plays the role of a server, in which it executes a development in Python so that only the images from the cameras are shown to the guard on duty, without him having access to manipulate or modify the cameras or the system in general.

**KEYWORDS:** Automatic opening, tensorflow, Python, facial recognition, video surveillance, computer vision

## **INTRODUCCIÓN**

La visión artificial o visión por computador es la ciencia que engloba distintas disciplinas tecnológicas y estadísticas como la electrónica, ciencias de la computación, estadística y matemáticas para enseñar a que las máquinas tengan la capacidad de aprender, al extraer la información de las imágenes digitales y resolver tareas mediante la interpretación de la escena que están visionando a través de varios procesos numéricos internos como por ejemplo pasar la imagen en 2D a 3D para intentar simular la visión humana.

Debido al incremento de la inseguridad en diferentes entidades escolares del país, se ha visto la necesidad de poseer un mecanismo que contraresta el problema social.

En el presente proyecto se mostrará la forma de acoplar procesos de redes neuronales y Machine Learning para un sistema de CCTV mediante visión por computador demostrando las maneras de vincular a otros campos de la ingeniería como la Electrónica o Redes en un entorno educativo.

## **ANTECEDENTES**

Uno de los pioneros de los sistemas de detección de personas es Pfinder (People Finder) (Wren et al., 1997), desarrollado en los laboratorios Media Labs del MIT (Massachusetts Institute of Technology), que emplea el modelo probabilístico Máximo A Posteriori (MAP) para detectar el cuerpo humano y realizar el seguimiento en planos 2D de imágenes. PFINDER demuestra la utilidad de características estocásticas, basada en la región para la interpretación de imágenes en tiempo real. Este enfoque permite interpretación interactiva y significativa de la forma humana sin hardware personalizado. La técnica es lo suficientemente estable como para soportar aplicaciones reales, así como técnicas de visión de orden superior.

El famoso “W4” (What, Where, When, Who) (Haritaoglu et al., 2000) desarrollado en la Universidad de Maryland, es capaz de detectar a varias personas en el exterior y analizar la silueta del cuerpo para inferir las actividades de las personas.

El VSAM (Video Surveillance And Monitoring) (Collins et al., 2000), fue desarrollado en la Universidad de Carnegie Mellon, el cual fue un proyecto de multicámaras de vigilancia en cooperación que fueron aplicados en el campus de la universidad.

Una investigación similar se ha llevado a cabo en los laboratorios de investigación privados de EE. UU.: en IBM, el grupo de People Vision Project (Connell et al., 2004) propuso nuevas soluciones para la detección basado en la apariencia, también en ambientes de interior desordenado.

En los laboratorios de Siemens, en el Departamento de Imágenes y Visualización [Comaniciu & Meer, 2002], se diseñó la primera formulación de seguimiento de personas basado en las técnicas de Mean-shift, y que siguen también a partes del cuerpo en ambientes de multitudes.

En Europa, desde 1996, el grupo del Prof. Blake en la Universidad de Oxford propone el enfoque CONDENSATION (CONDitional DENSity propaGATION) [Isard & Blake, 1998] para rastrear objetos en movimiento en cámaras con movimiento.

En el laboratorio de ImageLab en Italia, fue desarrollado el sistema SAKBOT (Statistical And Knowledge-Based Object Tracker) (Cucchiara et al., 2003) para detectar y rastrear han sido probadas para el seguimiento de las personas individuales a partir de una cámara personas y vehículos, utilizando un enfoque robusto a la oclusión y sombras. Se han utilizado en proyectos en colaboración con la Universidad de California en San Diego (Prati et al., 2003) para la seguridad y con las empresas europeas en el ámbito de los sistemas de transportes inteligentes (Cucchiara et al., 2005).

(Khan & Shah, 2003) en la Universidad de Florida del Sur propone un enfoque para el aprendizaje de información geométrica para el etiquetado consistente o etiquetado coherente espacial. Por ejemplo, para mantener la identificación de la persona y su trayectoria cuando él o ella se mueve desde el campo de visión de una cámara a la otra cámara, por medio de la reconstrucción geométrica homográfica.

En la actualidad varias empresas multinacionales como Panasonic, Google, Axxonsoft y Samsuns han desarrollado sistemas de reconocimiento facial como medios de seguridad y vigilancia para la identificación de usuarios y han obtenido excelentes resultados.

Ecuador adoptó la iniciativa de implementar un sistema de video vigilancia entre los 2010-2012 que formó parte del sistema integrado ECU 911 que permitió mejorar la forma de monitorear los alrededores de instituciones educativas y calles de mayor congestión, brindando una mejoría en la disminución del tiempo respuesta y actuación junto a la documentación

almacenada en materia de seguridad, así como el acceso a nuevos servicios, lo que apuntará al principio del buen vivir que se aspira en el país.

Por lo tanto, es de vital importancia continuar con el desarrollo de esta tecnología la cual ha entrado en apogeo en los últimos tiempos gracias al desarrollo de nuevos sistemas de software más rápidos y potentes.

## **PLANTEAMIENTO DEL PROBLEMA**

La seguridad de los niños y adolescentes en el país se ha convertido en un problema en los últimos tiempos debido a la presencia de redes de trata de personas que han promovido secuestros y robos. Desde el año 2012 la unificación de las guarderías, escuelas y colegios secundarios, han generado un aumento de estudiantes en los diferentes niveles escolares, lo cual ha incrementado los niveles de inseguridad en las entidades educativas por la reducida cantidad de personal destinada a la seguridad y vigilancia de los estudiantes.

La vigilancia en el Centro de Educación Inicial N. 1, se lo ha venido realizando a través del tiempo por el mismo cuerpo docente en los diferentes horarios establecidos. Luego bajo lineamientos institucionales se contrató una empresa de seguridad que provee de un guardia por turno para el control de más de cien estudiantes y padres de familia que asisten a la institución al ingreso y salida.

Al no existir suficiente personal de seguridad, se ha detectado que la puerta queda sin control y personas desconocidas ingresan a las instalaciones, lo cual ha provocado pérdidas de bienes y han puesto en peligro a los niños en los horarios de ingreso y salida. El control de apertura de la puerta principal se lo realiza mediante un pulsador que está al alcance de cualquier persona, lo cual no provee ninguna seguridad.

Debido a la ubicación del centro educativo y la falta de seguridad en áreas aledañas, se ha verificado que ingresan personas no autorizadas tanto en el día como en la noche, lo cual pone en riesgo al guardia de seguridad de turno, quien no está en la capacidad de responder en forma oportuna debido al tamaño de las instalaciones.



Por lo anteriormente expuesto, es necesario implementar un mecanismo que dé apoyo a las actividades de la unidad educativa, personal directivo, docente y de seguridad, con el fin de precautelar la integridad física de los educandos, incrementando el nivel en las instalaciones del plantel.

## **JUSTIFICACIÓN**

El proyecto de titulación se fundamenta en proveer un beneficio tecnológico para incrementar los niveles de seguridad y como consecuencia inmediata precautelar la integridad de los estudiantes, funcionarios y bienes del mismo, con el uso de las nuevas tecnologías y procesos que hoy en la actualidad se desarrollan en el ámbito de detección y reconocimiento mediante computadora.

El sistema de detección mediante cámaras de video vigilancia ubicadas en zonas estratégicas facilitará la detección de intrusos dentro de la institución y tomar las acciones correspondientes.

Gracias al sistema de reconocimiento facial existirá un mejor control en cuanto a las personas que ingresen a la institución. Las autoridades y personal de seguridad podrán mantener un registro de las actividades que se desarrollan y determinar la presencia de personas y acciones que atenten a la seguridad de los niños, el personal de la institución educativa y de las instalaciones, siendo este el beneficio social que aportará el proyecto.

El sistema de video vigilancia mediante visión por computador servirá como un instrumento para mejorar la seguridad del Centro de Educación Inicial 1 del Ministerio de Educación y se aprovechará como un modelo que podría ser implementado en otras instituciones educativas para proveer de medios tecnológicos de alto nivel que garanticen la seguridad de los niños y las instalaciones lo cual ayudará a tener mayor tranquilidad a padres, maestros y directivos.

## **OBJETIVO GENERAL**

Implementar un sistema de video vigilancia mediante algoritmos de Deep Learning y machine learning para el Centro de Educación No 1 del Ministerio de Educación.

## **OBJETIVOS ESPECÍFICO**

- Determinar la situación actual de seguridad del establecimiento
- Acoplar algoritmos de Deep Learning y machine learning para el reconocimiento facial del personal autorizado y la detección de personas.
- Definir y acoplar los elementos básicos que formarán parte del sistema de video vigilancia de reconocimiento facial y detección de personas.
- Validar la funcionalidad del sistema.

## **ALCANCE**

El presente proyecto de sistema de video vigilancia con visión por computadora contempla el estudio de las instalaciones de la institución educativa para determinar los requerimientos en cuanto a seguridad de las instalaciones.

Se realizará el diseño e implementación de un software para el reconocimiento de rostros acoplado a un mecanismo para la apertura automática de la puerta de acceso. Se implementará un sistema de detección de personas mediante cámaras digitales en zonas críticas de seguridad.

El software basado en Machine Learning y Deep Learning para la detección y reconocimiento de personas se desarrollará en software de licencia libre y formará parte del sistema que será entregado para su administración a las autoridades del Centro de Educación Inicial 1 del Ministerio de Educación una vez realizada la capacitación y pruebas de funcionamiento. Además de tener un respaldo de video vigilancia las 24 horas del día y los 365 días del año.

## **DESCRIPCIÓN DE CAPÍTULOS**

Capítulo 1.- Se presenta la fundamentación teórica sobre las componentes de video vigilancia, sus características técnicas, en especial a los conceptos fundamentales de Deep learnign, adicional se menciona cual es la realidad de los sistemas de video vigilancia en el Ecuador.

Capítulo 2.- Se realizó el marco metodológico sobre el desarrollo de otros temas relacionados con la aplicación de video vigilancia en base de investigaciones sobre el presente trabajo.

Capítulo 3.- En este capítulo se muestra la propuesta del sistema de video vigilancia por visión por computador y se identifica los elementos que componen el software con la elaboración de diagramas de flujo, el hardware necesario y la parte de las redes de conexión representadas con diagramas de bloque.

Capítulo 4.- Se desarrolla la propuesta, empezando el desarrollo de los lagorismos del programa de visión por computadora, circuito electrónico, el diagrama de objetos en cada una de sus etapas y las pruebas necesarias. Presenta las conclusiones y recomendaciones de la investigación realizada, finalmente se incluye con las fuentes bibliográficas.

# CAPÍTULO 1

## FUNDAMENTACIÓN TEÓRICA

### 1.1. SISTEMAS DE VIDEO VIGILANCIA

Actualmente el desarrollo de la electrónica, los sistemas de video, el almacenamiento y procesamiento digital ha provocado una amplia implementación de estos sistemas en zonas públicas y privadas. Estos sistemas son utilizados como medios de seguridad para proteger bienes y a las personas, es útil para identificar intrusos y personas que realicen actividades indebidas que ponga en riesgo la integridad de un lugar o individuo.

Un sistema de video vigilancia consiste en instalar cámaras de video con grabadores digitales que pueden ser monitoreados desde un monitor central en tiempo real, controlar diferentes dependencias y rincones, sin tener que desplazarse a cada sitio. (Intplus.es, 2019).

#### 1.1.1. Evolución de los sistemas de video vigilancia

El uso de las cámaras de seguridad se inicia en 1941 con el ejército alemán con cámaras en blanco y negro utilizadas para la observación de ensayos con misiles y la preparación de ataques a distancia. Se desarrollaron los generadores de cuadrantes para la observación de varias cámaras en una sola pantalla.

En 1949 se comercializan los primeros sistemas de circuito cerrado de televisión (CCTV) los cuales aún no disponían de sistema de grabación, en 1951 se desarrolló el primer sistema de almacenamiento mediante cintas magnéticas en formato VTR. Los primeros sistemas de vigilancia analógica trabajaban con cable coaxial y emitían señales de +0.5V y -0.5V lo que provocaba mucha interferencia y señales de mala calidad, toda la información es analógica, no se podían enviar datos de alerta. (Cat Colombia Solutions, 2016)

En 1986 Kodak desarrolló el primer sensor que daría paso a la imagen y video digital y para 1996 aparece la primera cámara IP, Neyete 200, a partir de este punto se desarrollaron los primeros sistema de seguridad digital basados en evolución informático y digital, apare la

grabación de video en alta calidad, unidades de almacenamiento digitales en disco duro y memorias, en 2002 emergen los primeros sistemas CCTV de reconocimiento facial y en 2005 el análisis de contenidos de video VCA. Para el año 2007, el 90% de la información de video se transporta a través de internet y en la actualidad la tecnología Turbo HD es el nuevo estándar de imagen de alta definición para cámaras y grabadoras de video. (Funcia, 2018)

### 1.1.2. Tipos de sistemas de video vigilancia

Los sistemas de video vigilancia se clasifican de acuerdo a la tecnología utilizada en sus sistemas de control, almacenamiento y el tipo de cámaras de video. Existen varios tipos de grabadores de video digital (DVR) disponibles en el mercado, a continuación, se presentan sus características: (Centronic, 2018)

- **CCTV Analógico:** Es el sistema más antiguo de CCTV, por sus características y buen rendimiento se continúan utilizando en ciertas aplicaciones, disponen de 4 a 32 canales tipo BNC, utilizan cámaras de tipo analógico, el DVR realiza la conversión digital para almacenamiento en disco duro.
- **CCTV IP Puro:** Este DVR soporta solo cámara IP, las cuales envían señales codificadas al grabador, el DVR tiene funciones de compresión de video MPEG4 H.264 y MJPE para almacenar la información en un disco duro. La conexión de las cámaras se realiza por cable UTP y telefónico.
- **Turbo HD 4.0:** Ofrece un sistema de muy alta compresión de video H.265+, 75% superior a sus antecesores y funciones inteligentes en tiempo real para la detección de objetos y personas, soporta cámaras de 8 Mega pixeles.

El centro educativo no dispone actualmente de un sistema de video vigilancia, por lo que se presenta como una mejor opción la implementación de un sistema de video vigilancia digital, debido a que en la actualidad son más económicos, sin embargo, en referencia al sistema Turbo HD 4.0, a pesar de tener tecnología de punta, su costo es un limitante ya que esta tecnología actualmente está empezando a ingresar al país.

En vista de que se necesita integrar el sistema de video vigilancia al de reconocimiento de rostros, se ha optado por una opción, que consiste en la adquisición de cámaras IP que puedan integrarse a la red a través de una red común que sea administrada por un computador que actuará como un sistema de CCTV IP puro, que se acoplará a las cámaras de video IP a través de una red, además se encargará de la compresión y del almacenamiento de video.

### 1.1.3. Cámaras de video

En el mercado existen muchos tipos de cámaras de video, por lo que es necesario identificar sus características, para establecer un criterio apropiado para su selección de acuerdo al lugar en que serán instaladas.

- **Cámaras de interior:** se muestra en la Figura 1.1, son las más sencillas, no requiere carcasa de protección ni visión nocturna si existe iluminación en las horas del monitoreo.



Figura 1.1 Cámara CCTV tipo mini bala para interiores

Fuente: (StereN, 2018)

- **Cámaras de exterior:** Su protección externa es mucho mayor ya que deben estar preparadas para soportar las inclemencias del tiempo como puede ser la lluvia, las bajas temperaturas o la suciedad, incluyen carcasas de protección y están catalogadas con un nivel de protección IP, que especifica su capacidad para soportar el agua y el polvo, lo cual puede afectar en el costo final de la cámara. Con nivel de protección IP66 puede soportar lluvia y temperaturas de  $-10^{\circ}\text{C}$  a  $50^{\circ}\text{C}$ , como se muestra en la Figura 1.2.



Figura 1.2 Cámara para exterior IP66

Fuente: (Hikvision, 2019)

- **Cámaras con infrarrojo:** Para lugares con poca iluminación o que requieren monitoreo nocturno, graban a color en el día y en escala de grises en la noche, en forma automática. La distancia que la cámara alcance a capturar en la noche depende de sus leds infrarrojos.
- **Cámaras antivandálicas:** Para zonas vulnerables a robos y agresiones, poseen carcasas resistentes a golpes como se muestra en la Figura 1.3.



**Figura 1.3** Cámara antivandálica fija para exterior

**Fuente:** (Panasonic.2019)

- **Cámaras IP:** Son sistemas completos que se conectan directamente a una red local o internet, pueden ser monitoreadas desde cualquier parte del mundo. Se observa en Figura 1.4



**Figura 1.4** Cámara CCTV IP

**Fuente:** (Serviplus, 2019)

- **Cámaras WiFi:** Son cámaras IP como se muestra en la Figura 1.5. de conexión inalámbrica, presentan facilidad para la instalación, pero pueden ser vulnerables a ataques con inhibidores de señal.



**Figura 1.5 Cámara WiFi**

**Fuente:** (Hikvision, 2019)

- **Cámaras con movimiento o zoom:** Son ideales para instalaciones que poseen personal para el monitoreo, cubren amplias áreas, enfocándose a una ruta en movimiento se puede ver en la Figura 1.6



**Figura 1.6 Cámara 10X zoom óptico**

**Fuente:** (Hikvision, 2019)

- **Cámaras ocultas:** Son cámaras espías que pasan totalmente desapercibidas, pueden ir instaladas en detectores de humo, interruptores, lámparas, espejos o como placas electrónicas dentro de prototipos elaborados en acrílico. se puede ver en la Figura 1.7





**Figura 1.7** Cámara oculta portátil WiFi

**Fuente:** (Amazon.ec, 2019)

- **Cámaras todo en uno:** Son sistemas completos que se conectan directamente a teléfonos celulares y tablets, almacenan las imágenes en tarjetas microSD internas. Dispositivo mostrado en la Figura 1.8.



**Figura 1.8** Cámara IP, 3G, 4G, WiFi, tarjeta SD, audio

**Fuente:** (Anitek, 2019)

#### 1.1.4. Características técnicas de una cámara de video vigilancia

La apropiada selección de una cámara de CCTV depende de una apropiada selección de sus características técnicas, las principales son las siguientes: (Junghanss, 2017)

- **Sensibilidad:** Es la cantidad mínima de luz que requiere la cámara para reproducir una imagen en bajas condiciones de luz, las cámaras a color pueden variar de 0.1 a 1 lux. como se muestra en la Figura 1.9. La medida en lux se refiere a una determinada apertura de óptica (F.1.2., F.1.4., etc) (Aceves, 2013)

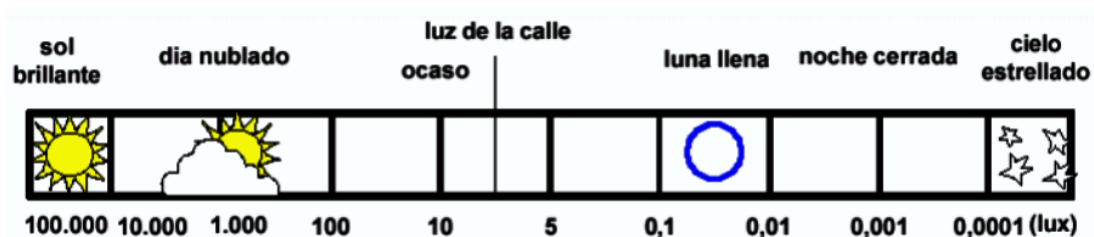


Figura 1.9 Rango de unidades de iluminación en Lux

Fuente: (Aceves, 2013)

- **Resolución:** Mostrada en la Figura 1.10 es la capacidad de la cámara para reproducir detalles finos, una cámara estándar tiene de 380 a 420 líneas de resolución (TVL). En base a los sensores de imagen CCD/CMOS, las cámaras analógicas pueden ofrecer vídeo de resolución estándar con 1000 TVL, 700TVL, 650TVL, 480, 420TVL para cámaras analógicas. En cuanto a cámaras digitales se emplea el número de píxeles para referirse a la resolución de imagen, existen cámaras desde 0,3 MP, 2 MP, 3MP, 5MP, 8MP y superiores, un mayor número de Píxeles representa una mejor calidad de imagen y una resolución más alta (Avitom, 2017). En los Kits CCTV (ya sean cámaras IP o cámaras analógicas), además de la calidad de imagen, también hay que tener en cuenta el tamaño de la misma: (Seguridad TV, 2018)



Figura 1.10 Tamaño de la imagen en función de MP

Fuente: (Seguridad TV, 2018)

La imagen resultante de una cámara digital es mejor que una cámara analógica como se observa en la Figura 1.11.



**Figura 1.11 Comparativa entre a) Cámara analógica de 600TVL y  
b) Cámara digital IP avanzada de 3MP**

**Fuente:** (Seguridad TV, 2018)

- **Iris electrónico:** Controla en forma automática la cantidad de luz que entra a la cámara, mediante el uso de un obturador electrónico automático, cuanto mayor sea la velocidad de control, mejor será la calidad de la imagen en condiciones de luz brillante, esta puede ir de 1/60 a 1/100.000 de segundo.
- **Ajuste de control de fase:** Permite sincronizar la frecuencia de la señal de video con la frecuencia de la red, evita saltos indeseables durante la reproducción del video en vivo o cuando se producen grabaciones después de un evento.
- **Relación señal/ruido:** Mide la inmunidad al ruido eléctrico proveniente de la red eléctrica, las normas recomiendan 46db como mínimo.

### **1.1.5. Estándar de compresión**

La compresión de video mostrado mediante Figura 1.12 consiste en reducir y eliminar datos redundantes del video digital para que pueda enviarse y almacenarse en forma eficiente. Para la codificación y decodificación del video se utiliza un algoritmo denominado códec de video.

El estándar de compresión y el formato de archivo deben ser apropiados para la transmisión del video, de modo de reducir el esfuerzo computacional y al mismo tiempo ofrecer la suficiente cantidad de cuadros por segundo, de modo que el video pueda ser utilizado como

evidencia ante un requerimiento jurídico. Se recomienda el uso de protocolos H.264 o MPEG4 que permiten un uso apropiado del ancho de banda con videos HD. (Aceves, 2013)



**Figura 1.12 Comparación tasa de compresión**

**Fuente:** (Security Set, 2016)

Se denomina latencia al tiempo necesario para comprimir, enviar, descomprimir y mostrar un archivo, cuanto más avanzado es el algoritmo de compresión, mayor será la latencia y por ende más eficiente el sistema de compresión.

H.264 se caracteriza por tener altas capacidades de predicción y recuperación ante errores, gracias a su flexibilidad ha sido empleado en aplicaciones de Blu-ray, difusión de video digital, televisión de alta definición, youtube, telefonía móvil de tercera generación, QuicTime, Flash, sistemas operativos como Apple y MacOS y Play Station 3. H.264 puede reducir el tamaño de video digital en más del 80% si se compara con JPEG y hasta un 50% comparado con MPEG4 versión 2. (Axis Communications, 2018)

#### **1.1.6. Sistemas de control de acceso**

Un sistema de control de acceso es un mecanismo que en función del reconocimiento ya autenticado permite acceder a un determinado recurso, existen múltiples formas de control de acceso, por ejemplo, controles de acceso por software que requieren digitar una contraseña, sistemas por huella dactilar, de reconocimiento facial entre otros. (Tecnoseguro, 2018).

El enfoque de seguridad electrónica de este trabajo se dirige a un sistema electrónico que permita el acceso de usuarios autorizados a un área específica en base a la validación de su identificación por medio de un tipo específico de lectura, en este caso el reconocimiento facial

y a su vez se encarga de la apertura de una puerta, por medio de un dispositivo electrónico (circuito de control y chapa eléctrica).

Las puertas abiertas y sin control representan un problema de seguridad y en muchos casos representan costos y pérdida de recursos por la necesidad de disponer de personal de seguridad en forma permanente en puntos específicos. Algunas ventajas de los sistemas de control de accesos son, acceso rápido de personas autorizadas a zonas restringidas, el control de ingreso y salida del personal mejora la productividad de la empresa, impide que personas ajenas a la organización ingresen, los sistemas electrónicos de seguridad revalorizan las instalaciones debido a la innovación y tecnología empleada.

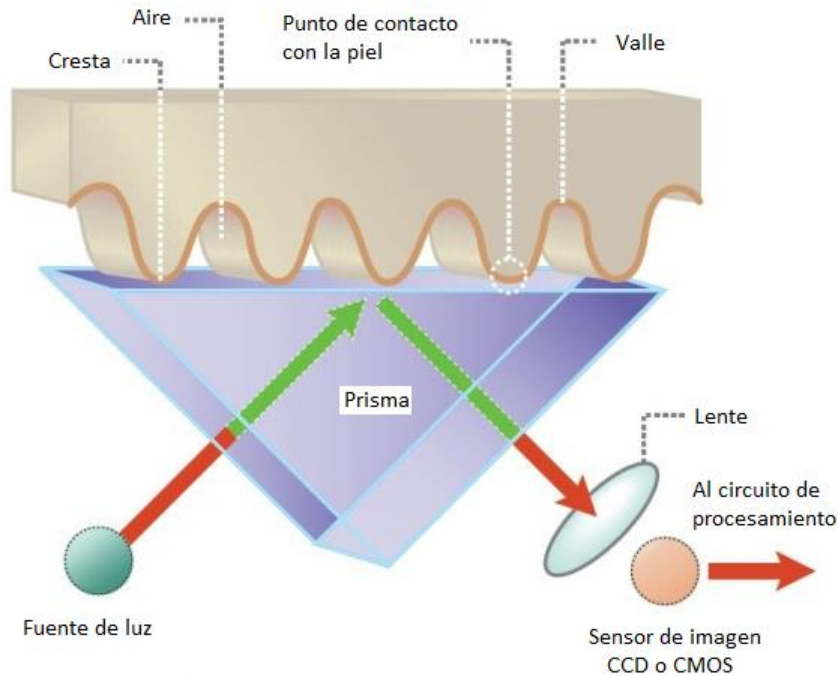
Existen dos tipos de sistemas de control de acceso:

- Los sistemas de control de acceso autónomos, que permite el control de puertas sin estar conectados a un computador o sistema central, por lo tanto, no guardan registros de entrada o salida, estos son muy sencillos y funcionan simplemente a través de una llave o tarjeta electrónica que acciona el dispositivo de bloqueo de la puerta.
- Los sistemas de control de acceso en red son sistemas complejos conectados hacia un computador o sistema centralizado cuyo software permite llevar un registro de las personas que entran y salen, permiten agregar datos como fecha, horario y las veces que ha empleado el sistema.

Los sistemas biométricos son aquellos que leen algún rasgo del cuerpo de la persona, como la huella dactilar, rasgos faciales o el iris del ojo, la ventaja de este sistema es que únicamente la persona autorizada puede accionar el dispositivo, de esta forma se incrementa el nivel de seguridad, a continuación, se presentan las principales características de estos sistemas:

- **Sistemas de acceso de huella dactilar.**

Es un dispositivo encargado de detectar los relieves del dedo por medio de luz o sensores electrónicos, en base a la cual se genera una imagen digital que es posteriormente almacenada y procesada. Las huellas digitales están formadas por pequeños valles y crestas de piel en la superficie de los dedos, cuya forma es única. Un lector de huellas digitales, obtener una imagen de huella y comparar la secuencia de valles y crestas con los patrones de valles y crestas que tiene almacenadas. Se puede observar mediante la Figura 1.13.



**Figura 1.13 Sensor óptico de huellas digitales**

**Fuente:** (Androide Libre, 2017)

- **Sistemas de acceso con control biométrico ocular.**

El reconocimiento del iris de la Figura 1.14. Emplea una cámara con una fuente de luz infrarroja que reduce el reflejo que se pueda producir en la convexa córnea, la cámara obtiene una imagen detallada del iris que es convertida en información digital. Una vez obtenida la imagen, esta es comparada en base a un algoritmo con la información obtenida previamente en una base de datos. Los ojos de las personas tienen patrones únicos e inequívocos, incluso el ojo izquierdo y el derecho de una misma persona presentan patrones distintos.



**Figura 1.14 Sensor óptico de huellas digitales**

**Fuente:** (EcuRed, 2019)

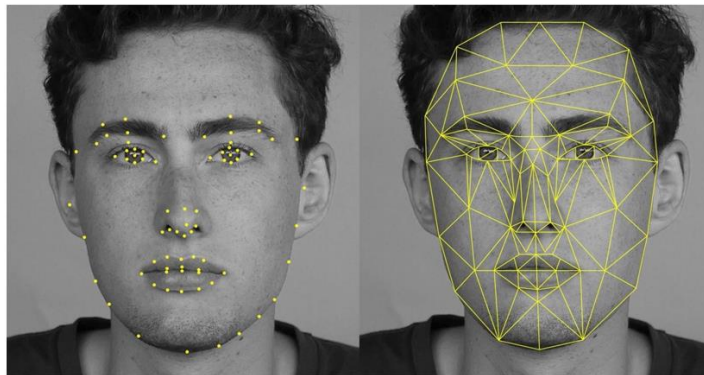
La identificación del individuo no es invasiva y no produce daño en el ojo, se ha demostrado que incluso con algunos tratamientos quirúrgicos su fina textura se mantiene inalterable, Este sistema es en extremo confiable para la identificación de personas y es invariable con el tiempo.

Un algoritmo de reconocimiento a través del iris fue patentado por Daugman en 1992. Wildes y Boashash & Boles han desarrollado otros sistemas, pero ninguno de ellos ha logrado el nivel de éxito y expansión comercial de Daugman. (EcuRed, 2019)

- **Sistemas de acceso de reconocimiento facial.**

En la Figura 1.15. Se ve el proceso de reconocimiento de las personas en bases a ciertos rasgos denominados características biométricas del rostro. La técnica se basa en sofisticados procesos matemáticos y algoritmos de coincidencia en base a un conjunto de imágenes denominadas imágenes de entrenamiento, el proceso se complica cuando se requiere que la identificación sea tiempo real.

Este es un método no intrusivo que requiere muy poca colaboración del individuo. Existen dos tipos de reconocimiento facial: los sistemas geométricos basados en la medición de rasgos (ojos, nariz, boca) y ángulos de la cara.



**Figura 1.15 Técnicas de reconocimiento facial**

**Fuente:** (Lacort, 2017)

Las huellas dactilares, el iris del ojo y el ADN son inalterables en la vida. En el reconocimiento facial factores como el envejecimiento, cirugía plástica, cosméticos, calidad de la imagen, efectos del consumo excesivo de drogas o tabaco, pose del sujeto, etc., pueden provocar detecciones erróneas en el sistema. (Vásquez, 2014)

Los métodos de reconocimiento de rostros se clasifican en dos grupos, los basados en características y los basados en la apariencia. Los métodos basados en características analizan las propiedades y la geometría del rostro, tales como áreas, distancias y ángulos entre los elementos del rostro. Los métodos basados en apariencias u holísticos realizan un análisis global mediante herramientas estadísticas. (Vásquez, 2014)

Según Esteban Vásquez (2017), responsable de Biometría en el área de Información Multimodal de Gradient, la biometría parte de la premisa de que ningún sistema biométrico es infalible, es decir, todos los sistemas tienen un porcentaje de error y ninguno lo tiene al 0%. Además, no se puede afirmar que el reconocimiento facial sea mejor "siempre" que el dactilar, o que la contraseña, o que la voz, o... Todos tienen ventajas e inconvenientes. El facial puede competir con el dactilar y ser superior, pero depende de su implementación.

Con esta premisa se optó por el sistema de reconocimiento facial como sistema de seguridad, para la implementación del presente proyecto.



## 1.2. VISION POR COMPUTADOR

Se define a la visión por computador al campo de la inteligencia artificial destinado a que una computadora pueda extraer información de una imagen mediante el uso de una cámara para adquirir, procesar, analizar, reconocer e interpretar imágenes para dar solución a problemas de la vida real. (González & Hernández, 2015). Consiste en un conjunto de técnicas para obtener, procesar y analizar imágenes y video para su posterior procesamiento en un computador, sus aplicaciones son cada vez más extensas para la toma de decisiones en las actividades para las cuales son programadas.

Determinar un camino libre de obstáculos, reconocer objetos del entorno, de entre ellos, posibles peligros y un largo etcétera, la visión por computador consiste en analizar imágenes mediante procesamiento matemático y digital para obtener información detallada de los objetos y sus características. Los datos de entrada son las imágenes y las peculiaridades de la imagen son la información de salida. La visión por computador no tiene como fin procesar imágenes digitales o la generación de gráficos por computador los cuales son campos afines.

Las ventajas o avances que presenta la visión por computador son bastante numerosas. A nadie se le oculta que dotar, en alguna medida, del sentido de la vista a los robots que se utilizan en la industria dispararía el número de aplicaciones. Otro ejemplo en el mundo industrial es el control visual de calidad ya que las cámaras permiten un control de calidad del 100% de lo que se fabrica. De hecho, estas dos aplicaciones son las que históricamente han desarrollado los algoritmos de análisis de imágenes y las cámaras digitales.

La llamada sociedad de la imagen lo es cada vez mayor, las cámaras se encuentran integradas en cualquier dispositivo electrónico. Se ha pasado de las cámaras en los ordenadores a los teléfonos, las tabletas, los televisores y ya se ven disponibles en las gafas. Ese aumento de cámaras ha propiciado encontrar imágenes en todas partes, con inmensas bases de datos de ellas, distribuidas de forma desorganizada por internet, almacenadas en los ordenadores personales, etcétera. También los sistemas de visión por computador son muy útiles a la hora de realizar gráficos por computador, a los que cada vez se les exige que sean más realistas.

Por todo ello el rango de aplicaciones de la visión por computador es cada vez más amplio. Existen aplicaciones en el campo militar, constituyen el núcleo central y tradicional, a las aplicaciones en el mundo de la seguridad, el control de tráfico, la identificación de personas que

son campos de aplicación emergentes y cada vez más importantes. Las tecnologías que han propiciado a que la visión por computador haya tenido un desarrollo tan grande los últimos años es el desarrollo de hardware. A nadie se le oculta que los microprocesadores cada vez son más potentes y de dimensiones más reducidas, con lo cual, por un lado, son capaces de procesar muchísima más información en menos tiempo, por otro lado, pueden ser incorporados en prácticamente cualquier dispositivo.

Esto es lo que ha llevado a que las cámaras sean introducidas en cualquier equipo electrónico y que en ese mismo dispositivo se puedan implementar algoritmos de análisis de imágenes. De nada serviría un gran hardware sin herramientas de software para procesamiento. Si además se tuviese siempre que empezar a partir de cero el diseño de aplicaciones informáticas estaría reducido a unas pocas empresas. Sin embargo, cada vez hay más herramientas informáticas de dominio público que permiten que el programador se centre en la implementación de la idea a partir de unos bloques básicos ya construidos denominados bibliotecas.

En el campo de la visión por computador destaca la biblioteca de funciones *OpenCV*, además de disponer de un buen hardware y herramientas informáticas, se necesita buenas ideas y capacidad para integrarlas en un sistema útil para el usuario final.

La inteligencia artificial también ha dado durante estos últimos años numerosos algoritmos que han permitido el análisis de imágenes, al salir del mundo de la industria a ambientes menos controlados. Un caso paradigmático son los programas de fotografía donde ahora en casi todos ellos se recuadra la cara de las personas que están presentes.

Hasta hace pocos años establecer qué características definían una cara en una imagen era una tarea no resuelta. Ahora, gracias a los algoritmos de inteligencia artificial, en concreto los de aprendizaje de máquina, esta aplicación está al alcance de todos.

### **1.3.1. Procesamiento de imágenes**

El procesamiento e interpretación de la imagen es lo que se conoce como procesamiento digital, la visión humana es uno de los sistemas de procesamiento de imagen más poderosos y complejos, aunque la mayoría del tiempo los seres humanos no le dan importancia a la cantidad de datos que son procesados a través de este órgano.

La visión es un proceso que se realiza en el subconsciente por lo que la persona no sabe cuáles son los pasos para llevarlos a cabo. Por lo tanto, tampoco conoce qué pasos deben de dar los computadores para analizar una imagen. En el caso de los ordenadores se presenta una complicación adicional ya que los computadores no captan imágenes como las que el ser humano ve, sino que para ellos constituye una matriz de números y las aplicaciones informáticas deben traducir esos números en objetos.

En el caso de aplicaciones industriales, donde el ambiente está controlado, esta tarea es más fácil que en el caso de aplicaciones en exteriores o en casa, donde existen muchas variables que no se pueden determinar o fijar por lo que la solución de las tareas se complica. Se puede dar el caso de que el objeto deseado no tenga una forma rígida, por ejemplo, un animal que presenta cambios debidos al movimiento.

Otro factor muy importante es la iluminación, que puede no ser constante y que no sólo va a producir que las imágenes se vean más claras u oscuras, sino que además producirán sombras, se modifica el contraste entre diversas zonas, etcétera.

En una aplicación normal el objeto buscado no va a ser el único que existe en la imagen. La presencia de otros objetos puede producir oclusiones con lo cual la búsqueda no será global, sino sólo de alguna de sus partes. Además, se complica la búsqueda ya que se presentan muchos posibles casos donde encontrarlo.

El procesamiento debe realizarse en pocos milisegundos, en ese poco tiempo se debe examinar decenas de miles de posibilidades en cada imagen, con lo cual, si el algoritmo funciona al 99,9% de acierto, se obtendrá también varias decenas de errores por cada imagen.

Estos errores serán falsos positivos, cuando se afirma que los objetos buscados están presentes en la imagen y no es así, o falsos negativos, cuando al estar presente la imagen, el sistema no lo reconoce. Son estas dificultades las que hacen que implementar un algoritmo de visión por computador sea una tarea compleja y apasionante.

Las técnicas de procesamiento de imágenes tienen como fin realzar o modificar imágenes para mejorar su apariencia, para medir, contrastar o clasificar algún elemento de la imagen para destacar algún aspecto de la información. (Mosquera & Romero, 2016). Los objetivos principales que se busca en el procesamiento de imagen son: (Querejeta, 2015)

- Suavizar la imagen: reducir cambios de intensidad entre píxeles vecinos.
- Eliminar ruido: eliminar píxeles cuyo nivel de intensidad es muy diferente al de sus vecinos.
- Realzar y detectar bordes: detectar grupos de píxeles con cambios bruscos de intensidad.

### **1.3.2. Etapas del procesamiento de imágenes**

Una imagen digital es una función  $f(x,y)$  que se discretiza en filas y columnas y a cada posición se le ha asignado un nivel de brillo, así se conforma una matriz de elementos denominados pixels o pels. La imagen digital definida como la representación numérica de un objeto debe ser sometida a una serie de operaciones organizadas en etapas para obtener los resultados deseados. Estas etapas están organizadas dentro de tres niveles de procesamiento: (Mut & Romero, 2016)

- **Visión de bajo nivel**

Procesos automáticos, no requieren ningún tipo de inteligencia, este nivel contiene las actividades de Adquisición y la de pre procesado.

- **Visión de nivel intermedio**

Está asociada a los procesos que extraen, caracterizan y etiquetan componentes de la imagen que se obtiene de la visión de bajo nivel. Involucra los procesos de Segmentación, Representación y descripción (Querejeta, 2015).

- **Visión de alto nivel**

Se refiere al procesamiento cognitivo, basados en el conocimiento y la comprensión de los procesos de visión, son más difusos y especulativos. Estos procesos se denominan reconocimiento e interpretación.

La Figura 1.16 presenta el procesamiento de imágenes subdividido en sus actividades principales: Adquisición, Procesado, Segmentación, Representación y descripción y Reconocimiento e interpretación (Mut & Romero, 2016).

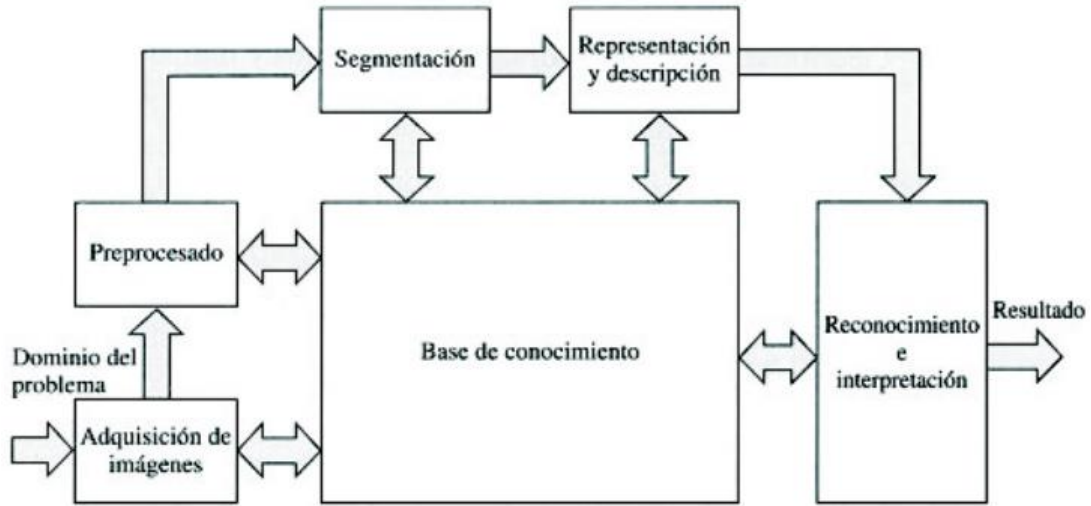


Figura 1.16 Etapas del procesamiento de imágenes

Fuente: (Mut & Romero, 2016)

- **Adquisición de imágenes.**

La Adquisición de las imágenes como por ejemplo las que muestra la Figura 1.17 es el proceso por el cual se obtiene la imagen, se necesita un sensor de imágenes y la posibilidad de convertirla en píxeles.

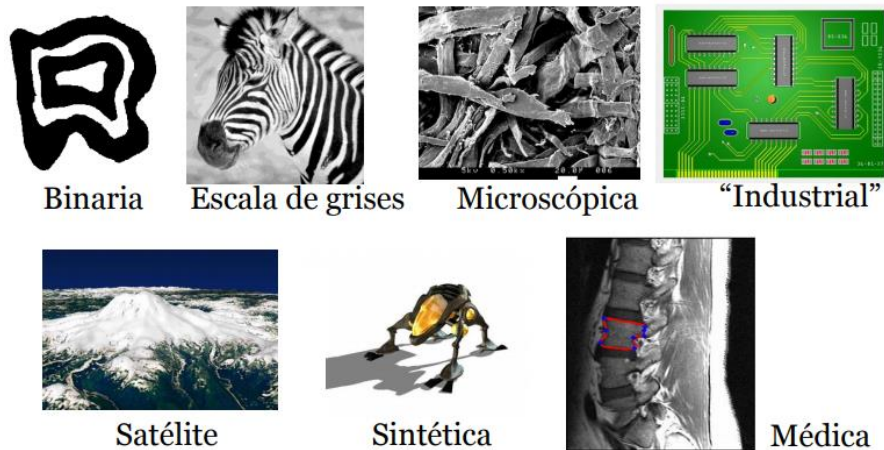


Figura 1.17 Tipos de imagen para procesamiento

Fuente: (Jimenez, 2018)

- **Pre procesado.**

El pre procesado consiste en la reducción del ruido, suavizado, realce de detalles, mejora de contraste, al binarizar imágenes y técnicas de detección de bordes, permite aumentar la probabilidad de éxito en los procesos posteriores. Se puede observar en la Figura 1.18

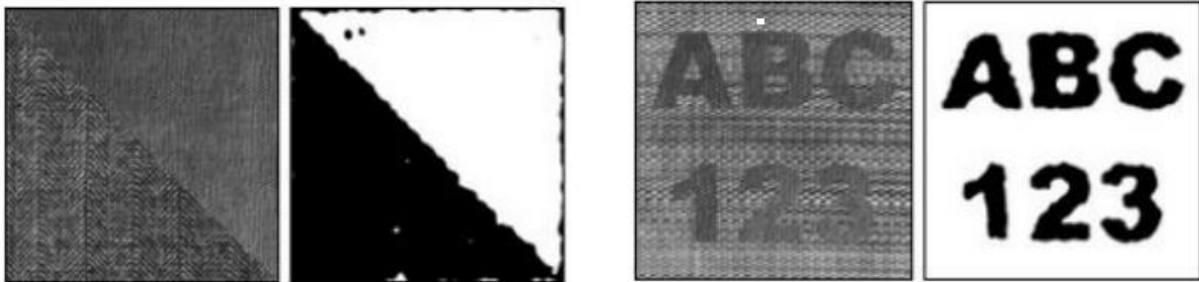


**Figura 1.18 Eliminación de ruido**

Fuente: (Davies, 2012)

- **Segmentación**

La Segmentación como lo muestra la Figura 1.19 consiste en dividir la imagen en regiones significativas de características homogéneas distintivas, denominadas regiones de interés o ROI, pueden ser tan distintas como la identificación de una fruta madura o el rostro de una persona (Jimenez, 2018).



**Figura 1.19 División de imágenes en regiones**

Fuente: (Jimenez, 2018)

- **Representación y descripción**

Del proceso de segmentación se obtienen grupos de píxeles que forman el contorno de una región, los puntos de una región o una combinación de ambos. Como se observa en la Figura 1.20.

La representación trata a la imagen para obtener el contorno de la imagen si el interés es obtener la forma, o la región completa si el interés se centra en la estructura o la textura del

objeto. La descripción consiste en extraer rasgos con alguna información cuantitativa de interés o que sean de importancia para diferenciar un objeto de otro (Jimenez, 2018).

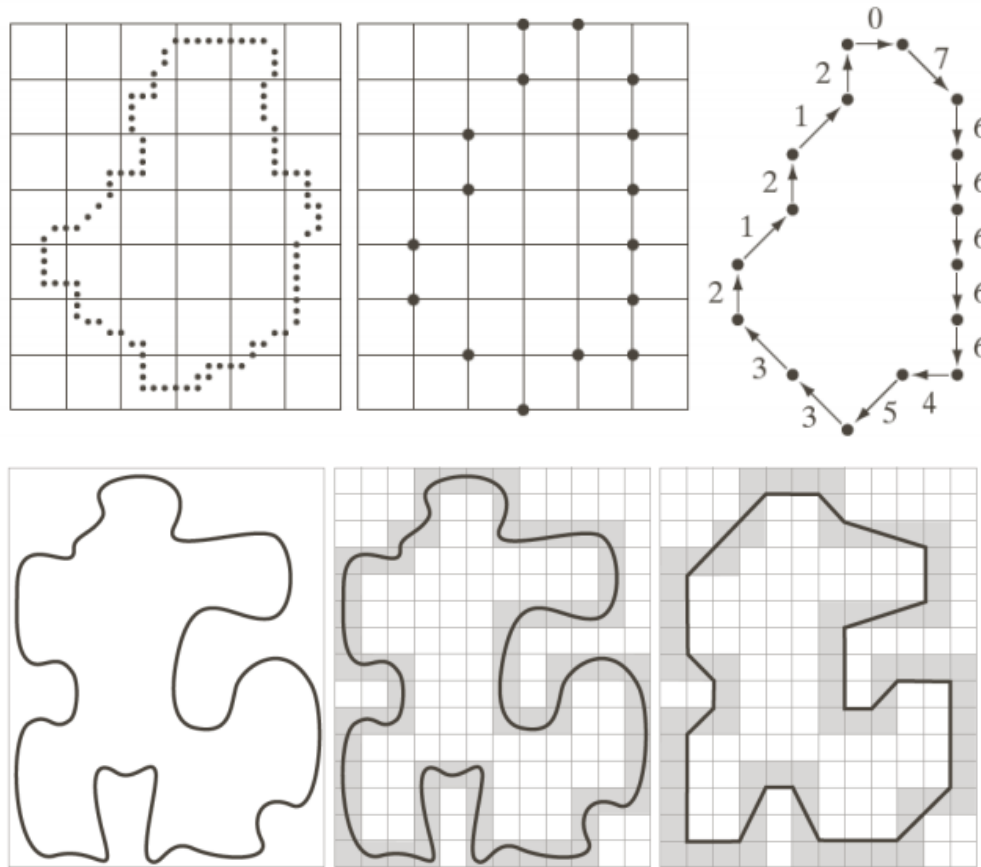


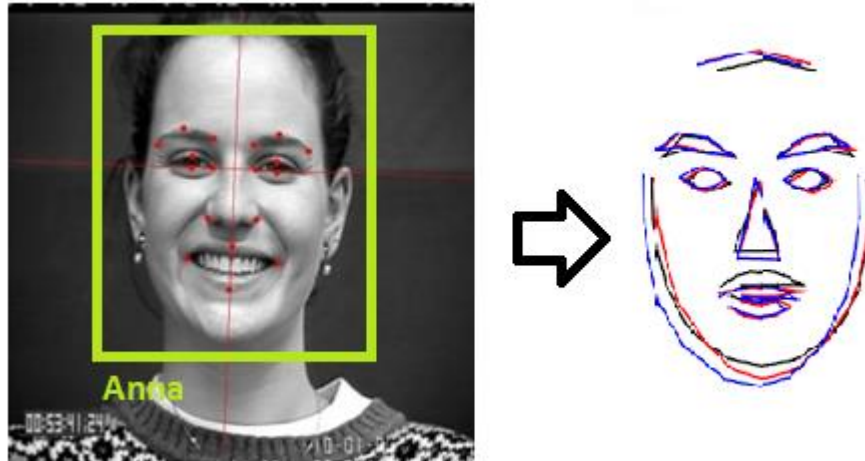
Figura 1.20 Obtención de contornos y rasgos característicos

Fuente: (Jimenez, 2018)

- **Reconocimiento e interpretación**

El reconocimiento consiste en asignar etiquetas a los objetos de acuerdo con la información provista por sus descriptores. La interpretación asigna una identificación a un conjunto de objetos reconocidos.

Esta etapa del análisis emplea algoritmos de aprendizaje de máquina que se muestra en la Figura 1.21 y con funciones complejas de reconocimiento basadas en inteligencia artificial, Machine Learning, Deep Learning, algoritmos complejos de agrupación, regresión, reducción y clasificación, redes neuronales y convolucionales para el análisis matemático de grandes cantidades de información.



**Figura 1.21 Reconocimiento digital de rostro mediante algoritmos de aprendizaje de máquina**

**Fuente:** Autor

### **1.3. INTELIGENCIA ARTIFICIAL**

La inteligencia es un concepto amplio y exhaustivo, puede definirse como la capacidad de un agente para alcanzar objetivos en una amplia gama de entornos; la dificultad para captar lo que es en realidad la inteligencia se traslada directamente a los intentos de emularla en las máquinas. Dentro del ámbito de las máquinas el término inteligencia artificial (IA) se limita a la capacidad de un agente para lograr sus objetivos en una gama (muy) limitada de entornos. (Linde & Schweizer, 2019). En todo caso, es una máquina que puede imitar comportamientos inteligentes.

En la actualidad la inteligencia artificial se encuentra presente en un campo amplio de aplicaciones que se emplean a diario. Como por ejemplo se muestra en la Figura 1.22. Debido a cada vez se acercan más al desarrollo de destrezas y comportamientos humanos, estas pasan desapercibidas por los usuarios, desde sistemas que hallan información en internet mediante instrucciones de voz, proveen seguridad integral, hasta aquellos que conocen las preferencias de los usuarios al seleccionar un programa de televisión digital.



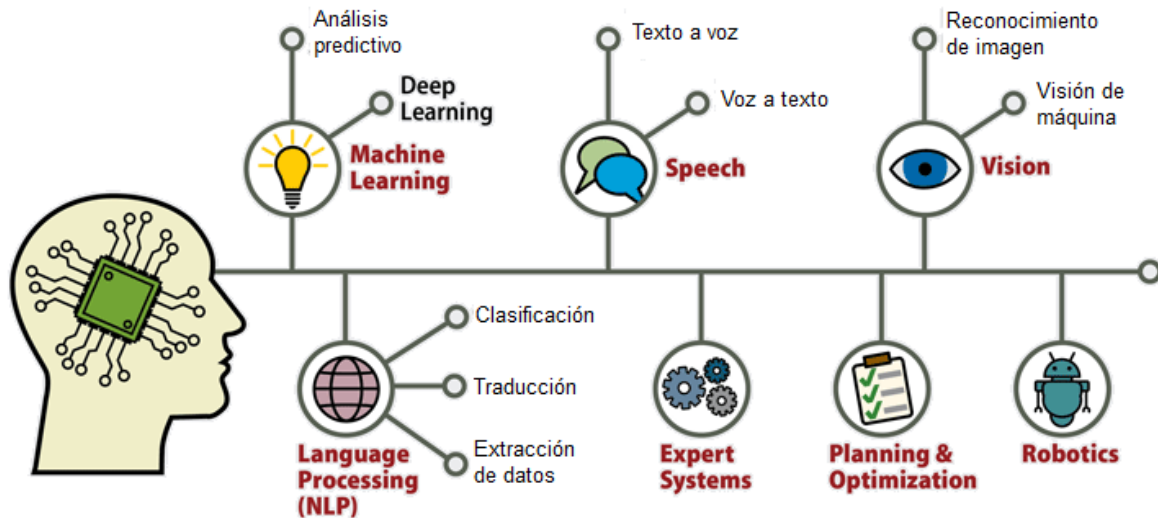


Figura 1.22 Aplicaciones de la inteligencia artificial

Fuente: (Datamation, 2019)

La IA ha evolucionado considerablemente desde los años 50 en que John McCarthy fijó los primeros parámetros y habilidades que debía poseer una máquina para ser considerada inteligente, en base a la prueba de Alan Turing (González R. , 2007), que ponía a prueba la habilidad de una máquina para hacerse pasar por un ser humano. Según Stuart Russel y Peter Norvig, en la actualidad puede establecerse las siguientes categorías de IA: (Andalucía es digital, 2018)

- **Sistemas que piensan como humanos.** Procuran imitar el pensamiento humano, las redes neuronales artificiales emulan la toma de decisiones, se los emplea en la resolución de problemas que requieren análisis y el aprendizaje autónomo.
- **Sistemas que actúan como humanos.** Sistemas informáticos que imitan el comportamiento de las personas o realizan movimientos similares.
- **Sistemas que piensan racionalmente.** Tratan de emular formas de pensamiento racionales que relacionan objetos con situaciones abstractas.
- **Sistemas que actúan racionalmente.** Sistemas que tratan de emular de forma racional el comportamiento humano, emplean razonamiento y pueden ser utilizadas en la toma de decisiones en base al análisis de millones de datos estadísticos o condiciones previamente establecidas.

### 1.3.3. Machine Learning

Machine Learning (ML), traducido al español como aprendizaje automático, es una de las ramas de la inteligencia artificial que evoluciona hacia los sistemas que piensan como humanos. ML busca resolver cómo construir programas de computadora que mejoran automáticamente a partir de un set de datos, básicamente esta tecnología detecta patrones de comportamiento. Los algoritmos de Machine Learning se dividen en tres grupos:

- **Algoritmos ML de aprendizaje supervisado.** Involucran la supervisión directa de la operación, el desarrollador etiqueta los datos de muestra (labeled data) e intenta encontrar una función que discrimine un dato de entrada y le asigne una etiqueta de salida adecuada. El algoritmo mostrado en la Figura 1.23 se entrena con un histórico de datos y así aprende a asignar las etiquetas a los datos de salida. (Recuero, 2017)

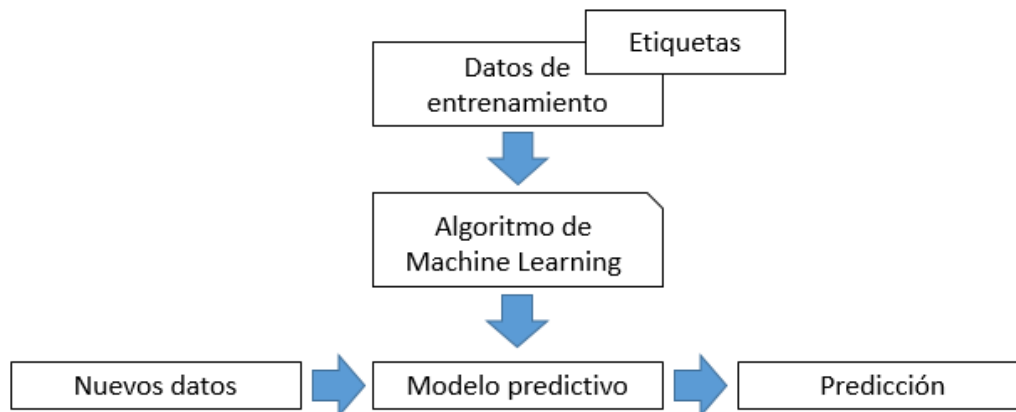
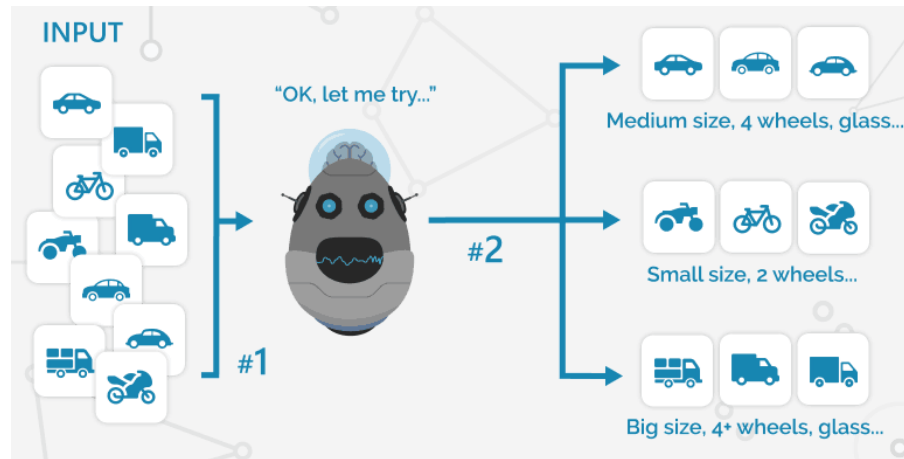


Figura 1.23 Algoritmo de aprendizaje supervisado

Fuente: Autor

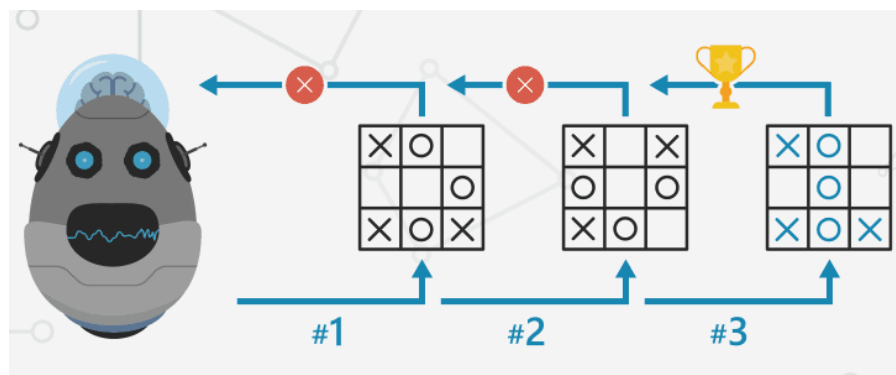
- **Algoritmos ML de aprendizaje no supervisado.** Similar al anterior, este sistema segrega datos no etiquetados en base a alguna característica en común de los datos. Es útil para descubrir estructuras ocultas en los datos, como en la detección de anomalías. A continuación se muestra en la Figura 1.24.



**Figura 1.24** Algoritmo de aprendizaje no supervisado

**Fuente:** (Chas, 2019)

- **Algoritmos ML de aprendizaje por refuerzo.** Consiste en la iteración constante y basada en “prueba y error” el procesamiento se realiza en muy poco tiempo cuando la máquina conoce de antemano el proceso, las condiciones de operación y se establece una meta o recompensa. En la Figura 1.25 el procesador ejecuta infino número de posibilidades del juego de las cuales va aprendiendo, mientras las condiciones para ganar son la recompensa.

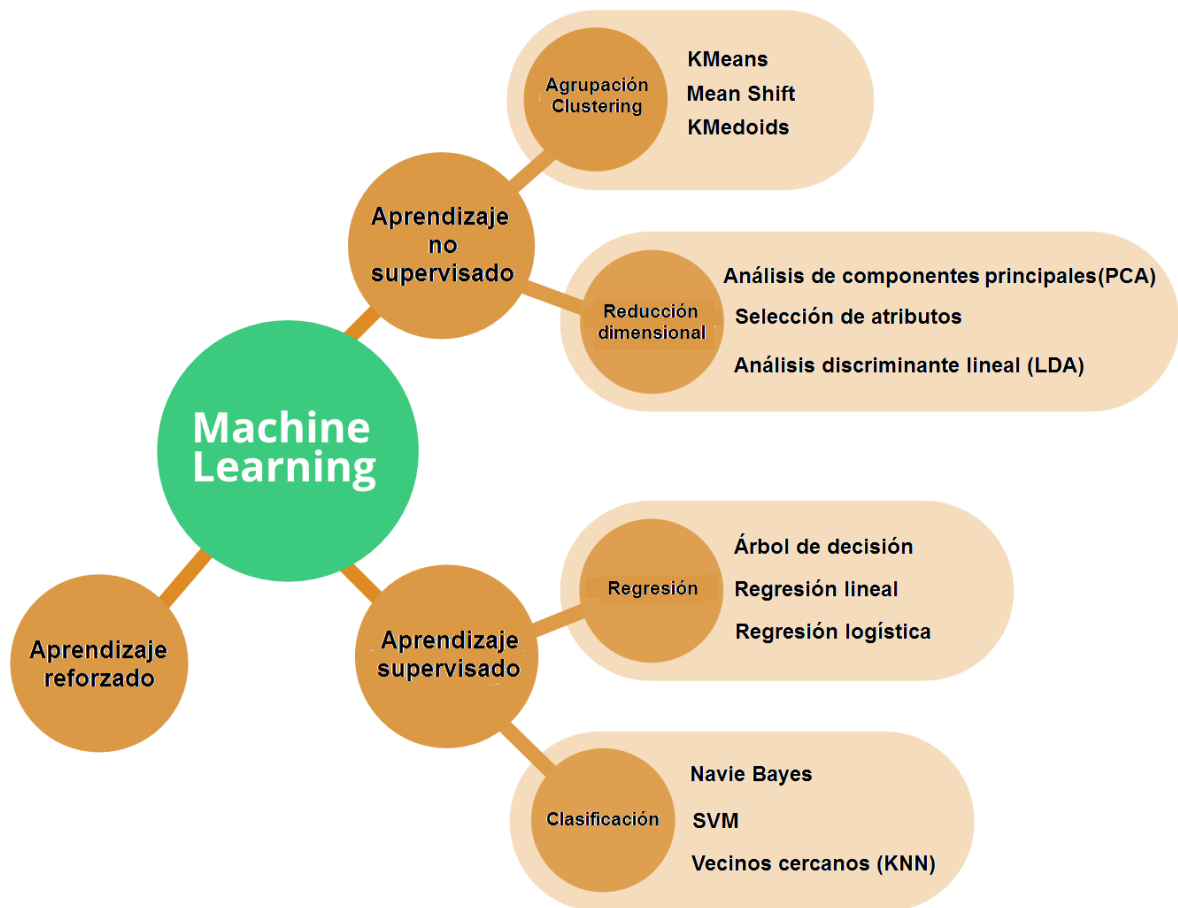


**Figura 1.25** Aprendizaje de un juego mediante algoritmo ML por refuerzo

**Fuente:** (Chas, 2019)

Existen múltiples aplicaciones que aprovechan esta capacidad de aprender y anticipar comportamientos de ML, desde el reconocimiento facial, diagnósticos médicos y sistemas de vehículos autónomos. Para desarrollar estas aplicaciones existen múltiples

algoritmos de ML que pueden ser reutilizados para la resolución de problemas, los mismos que se detallan en la Figura 1.26: (Bagnato, 2017) .



**Figura 1.26 Clasificación de los algoritmos de Machine Learning**

**Fuente:** (Vernon, 1991)

- **Algoritmos de agrupación Clustering.** Agrupa datos de características en común que no se conocen o se desea conocer. Este método intenta crear puntos centrales y jerarquías para diferenciar grupos y descubrir características comunes por cercanía (Vernon, 1991).
- **Algoritmos de reducción dimensional.** Reducen el número de variables para trabajar solo con la información exacta requerida, algunas razones para reducir la dimensionalidad son: reducir la complejidad, identificar y eliminar variables irrelevantes, mejorar el rendimiento computacional.

- **Algoritmos de regresión.** Modelan la relación entre distintas variables *features* que utilizan una medida de error que se intentará minimizar, a este proceso se lo denomina optimización y se ejecuta en base a un gran número de iteraciones.
- **Algoritmos de clasificación.** Consiste en predecir las etiquetas de clase categóricas de nuevas instancias en base a observaciones previas, pueden ser de dos tipos. El algoritmo encuentra la relación entre las características de cada punto de datos y su clase, y los separa mediante líneas.

Para elegir el mejor algoritmo de *Machine Learning* es necesario conocer el tipo de datos que se dispone y establecer qué se desea obtener como respuesta, lo cual permitirá escoger el algoritmo que mejor se adapte a las necesidades. En muchos casos se debe aplicar varios algoritmos a la vez para alcanzar los resultados deseados.

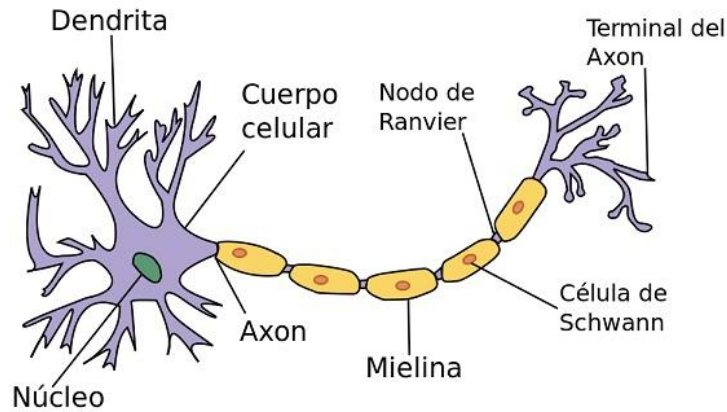
## 1.4. DEEP LEARNING

El concepto de *Deep Learning* apareció de la tesis doctoral de Paul Werbos en 1974, quien fue el primero en mencionar el proceso de entrenamiento de una red neuronal artificial mediante retro propagación. Se caracteriza porque para su funcionamiento no requiere de reglas programadas sino que el propio sistema es capaz de aprender por sí mismo basado en un entrenamiento previo. (Smart Panel, 2019)

Deep Learning requiere de una gran cantidad de datos etiquetados ya que está basado en aprendizaje supervisado, además necesita de procesadores de alto rendimiento ya que las redes neuronales emplean arquitectura paralela, es decir realizan múltiples cálculos al mismo tiempo.

### 1.4.1. Redes neuronales

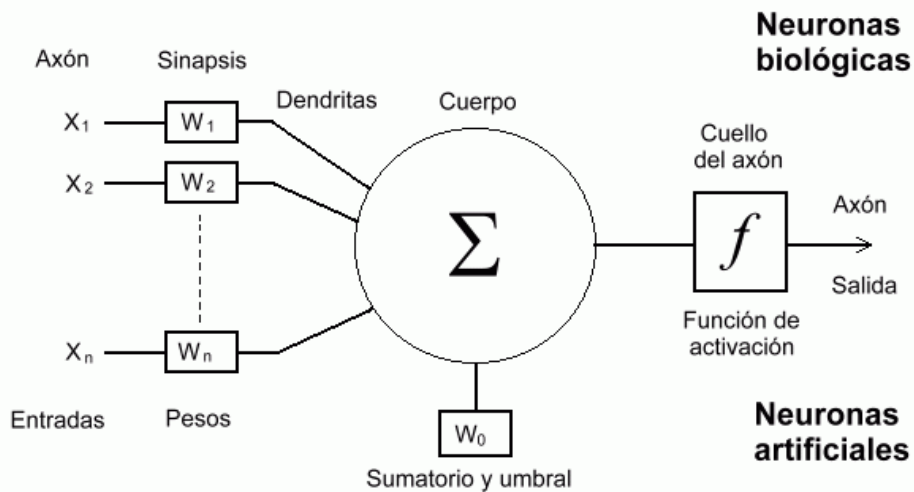
Los sistemas neuronales son una abstracción de los sistemas nerviosos biológicos como muestra la Figura 1.27 constituidos por neuronas y conexiones entre ellas denominadas sinapsis (mecanismos de comunicación entre neuronas), los cuales son la clave para el procesamiento de la información. Una neurona está conformada por un cuerpo celular, dendritas y axón (López M. , 2017)



**Figura 1.27 Ejemplo de una red neuronal**

**Fuente:** (López M. , 2017)

La forma en que dos neuronas se relacionan depende de su tipo, en general la respuesta de una neurona se envía a través de señales analógicas emitidas por su axón, la respuesta depende de las señales obtenidas por sus dendritas debido a su sinapsis, si la relación es positiva, la respuesta será positiva. Una neurona artificial ha sido diseñada con elementos similares a los de una neurona biológica y se representa en la siguiente Figura 1.28.



**Figura 1.28 Comparación entre una neurona biológica y una artificial**

**Fuente:** (Sancho, 2009)


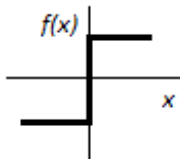
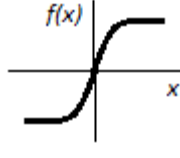
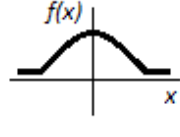
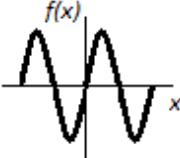
Una neurona artificial posee un grupo de entradas  $X_n$  con sus correspondientes pesos sinápticos  $W_n$ , una entrada independiente denominada sumatorio y umbral  $W_0$ , un cuerpo que

realiza una operación sumatoria  $\Sigma$ , conectada a una función de activación  $f$  y a una salida. La respuesta es:

$$y = f\left(\sum_{i=1}^n w_i x_i\right) = f(u(x)) \quad (1)$$

La neurona puede adaptarse al medio circundante y aprender de él al modificar el valor de sus pesos sinápticos y por ello son conocidos como los parámetros libres del modelo (López M. , 2017). La función de activación  $f$  a la salida depende de la tarea a realizar y puede ser alguna de las presentadas en la Tabla 1.1.

**Tabla 1.1 Funciones de activación neuronal.**

	Función	Rango	Gráfica
Identidad	$y = x$	$[-\infty, +\infty]$	
Escalón	$y = \text{sign}(x)$ $y = H(x)$	$\{-1, +1\}$ $\{0, +1\}$	
Sigmoide	$y = \frac{1}{1 + e^{-x}}$ $y = \tanh(x)$	$[0, +1]$ $[-1, +1]$	
Gaussiana	$y = A e^{-Bx^2}$	$[0, +1]$	
Sinusoidal	$y = A \text{sen}(\omega x + \varphi)$	$[-1, +1]$	

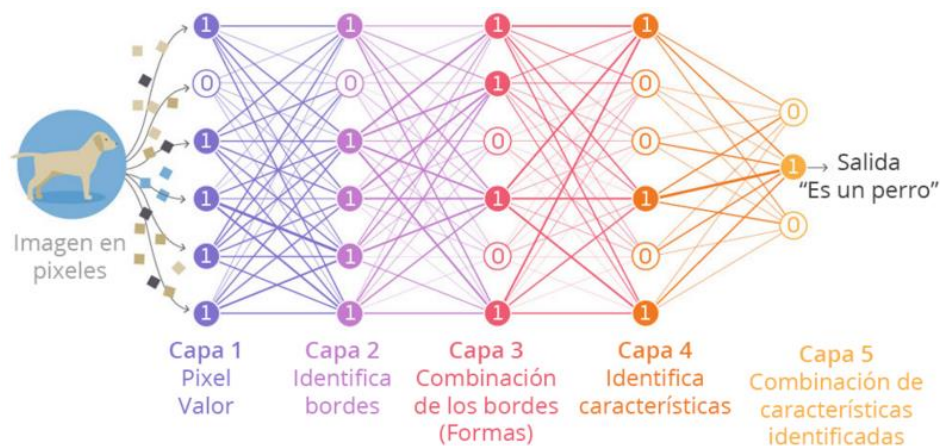
Fuente: (López M. , 2017)

### 1.4.2. Redes neuronales multicapa

Deep Learning es un subconjunto de aprendizaje automático derivado de Machine Learning, el cual se basa en redes neuronales de múltiples capas que aprenden en base a grandes cantidades de datos. Deep Learning se caracteriza por su estructura y procesamiento similar a las redes neuronales del cerebro humano.

En la actualidad Deep Learning ha permitido el desarrollo de aplicaciones de diagnóstico médico, predicciones en la bolsa de valores, traductores inteligentes como Google Translate y DeepL, procesamiento y reconocimiento de la voz como Siri, asistente de Apple y Alexa, dispositivo inteligente para el hogar de Amazon. Otra aplicación es el reconocimiento de rostros y gestos en tiempo real, empleados como llaves de seguridad en teléfonos celulares y la identificación de delincuentes en aeropuertos.

La siguiente es una red neuronal diseñada para identificar una imagen analógica, la cual debe ser transformada a información digital para introducirla en la capa de entrada, la imagen queda transformada en un grupo de píxeles que son enviados a las neuronas de la siguiente capa. La segunda capa delimita los bordes de la figura, la tercera combina los bordes para diseñar las formas, la cuarta emplea filtros para identificar los componentes de la imagen y finalmente la quinta capa recibe un grupo de fragmentos de información que le dicen, si existe una cola, cuatro patas, orejas, etc., y al combinar los datos presenta el resultado de si es o no un perro. Proceso que puede ser visualizado en la Figura 1.29.



**Figura 1.29** Ejemplo de una red neuronal

**Fuente:** (Smart Panel, 2019)



Los algoritmos basados en Deep Learning tienen en teoría la capacidad de optimizar cualquier función, en la realidad el sistema requiere de mucho procesamiento para intentar alcanzar un nivel de configuración óptima.

En los últimos años las redes Deep learning se han puesto en auge debido a las mejoras en el software de programación y el desarrollo de procesadores más potentes. Estas redes están formadas por millones de neuronas artificiales simples interconectadas, con capacidad de aprender reglas y condiciones.

Deep Learning funciona mediante la reducción de errores, cada capa se encarga de mejorar la probabilidad de acierto hasta 100%. Para que la maquina aprenda, se requieren dos etapas de aprendizaje, en la primera se emplea el aprendizaje supervisado, donde el programador etiqueta a la imagen para su identificación. En la segunda etapa se aplica aprendizaje no supervisado, donde el procesador halla sus propios patrones de aprendizaje y crea relaciones en base a datos previamente aprendidos.

En una red multicapa el problema radica en asignar valores adecuados a los pesos de cada neurona. Si  $w_{ji}$  es el peso sináptico entre una neurona de entrada  $i$  y una neurona oculta  $j$ , la entrada que recibe la neurona de la capa oculta será: (López M. , 2017)

$$u(x_i) = \sum_{i=1}^n w_{ji}x_i + \theta_j \quad (2)$$

Donde  $\theta_j$  es el umbral o sesgo de la neurona y su señal de salida será una función de la señal de entrada:

$$y_j = f(u(x_i)) \quad (3)$$

En la etapa de aprendizaje de la red neuronal, se calculan los pesos sinápticos de cada neurona al inicio a partir de datos aleatorios y posteriormente en base a iteraciones y datos de prueba. En ocasiones la red no aprende por que los datos son insuficientes o por no estar acorde al comportamiento deseado. Los pesos se van autoajustan hasta que se reduzca el error a la salida a valor mínimo.

### 1.4.3. Redes neuronales convolucionales

La convolución es una operación matemática que emplea dos funciones que son integradas previo al desplazamiento de una de ellas en el tiempo.

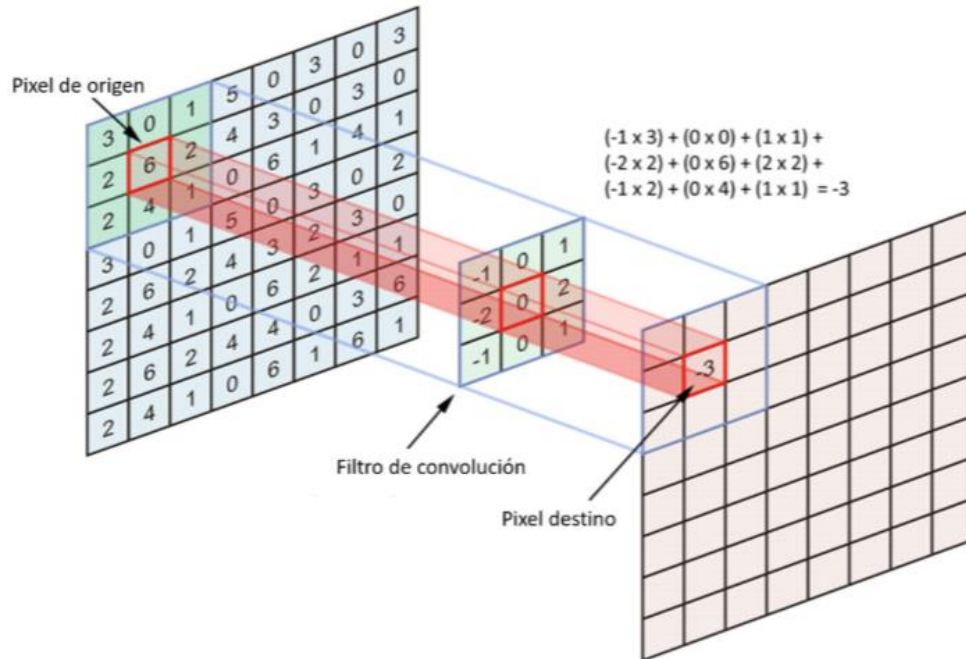
$$f * g = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (4)$$

Donde f y g, son las funciones,  $\tau$  es el tiempo en que se presentan las funciones y t es la distancia desplazada de la segunda función. En el dominio discreto la integral puede reemplazarse con una suma, de la cual se obtiene la siguiente ecuación:

$$f * g = \sum_{-\infty}^{\infty} f(k)g(n - k) \quad (5)$$

Donde, la función g está desplazada una distancia n.

La convolución puede ser representada como una operación matricial en la cual una matriz denominada deslizante se desplaza sobre otra a la vez que multiplica cada uno de sus elementos con los de la matriz, luego los suma para obtener el elemento central de matriz como salida, así se observa en la Figura 1.30.



**Figura 1.30 Proceso de convolución bit a bit**

**Fuente:** (López M. , 2017)

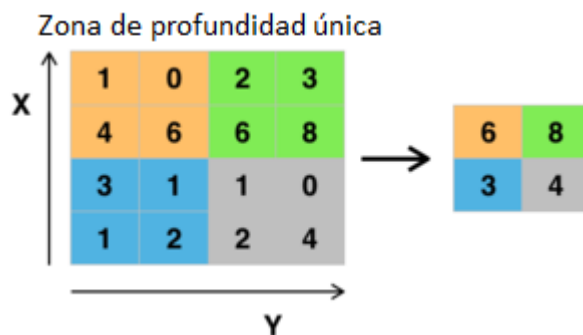
La matriz deslizante se comporta como un filtro que se desplaza a lo largo de toda la matriz, cambia las características de la imagen, tales como, reducción de ruido de alta o baja, difumina, enmarca, suprime señales no deseadas o separa señales en partes. La función realizada depende de los valores del filtro de convolución.

Las redes neuronales convolucionales (CNN) son redes neuronales especializadas que incluyen conocimiento sobre la invariancia de formas bidimensionales, utiliza patrones de conexión local y con restricciones en los pesos (LeCun & Boser, 1989), han sido desarrolladas en base al funcionamiento del cortex del ojo humano, el cual tiene la capacidad de distinguir distintas características de un objeto. Las CNN tienen varias capas ocultas especializadas y con jerarquía, las primeras capas reconocen líneas, mientras las más internas especializadas son capaces de reconocer rostros y siluetas de animales. (Bagnato, 2017)

Las CNN se diferencian de las redes neuronales porque han sido diseñadas para el procesamiento de imagen de fotografía y video. La CNN está conformada por una capa de entrada que recibe la información en pixeles, la información pasa a las capas ocultas donde se realizan dos operaciones.

La primera operación es la convolución, como ya se analizó anteriormente, el filtro convolucional posee una pequeña ventana que barre toda la imagen original y entrega un nuevo tipo de imagen del mismo tamaño en ancho y altura de la original, pero se establecen varios filtros de interés como por ejemplo sombras, contrastes, bordes, etc. Como cada uno de ellos generará una nueva imagen, la profundidad de imagen se incrementa de acuerdo con el número de filtros.

La segunda operación es el Pooling, consiste en la reducción de las dimensiones espaciales de la matriz entregada por la operación de convolución, tiene como fin reducir la sobrecarga de cálculos y el sobreajuste producido cuando el sistema almacena una exagerada cantidad de información sin generalizar las reglas para predecir los nuevos datos. La operación de Max-pooling toma grupos de pixeles vecinos de cuatro en cuatro y los reemplaza por el máximo valor del grupo de pixeles como se observa en la Figura 1.31, de esta forma, puede reducir el número de pixeles a la mitad.

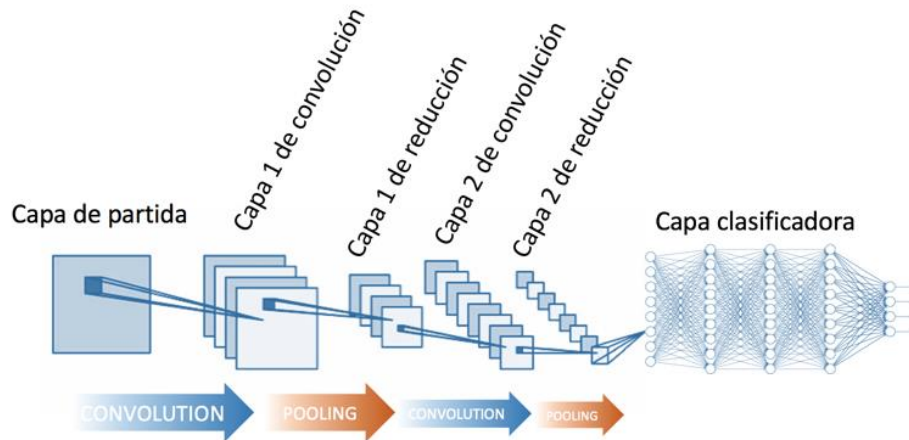


**Figura 1.31** Proceso de pooling en una matriz 4x4, resultante 2x2

Fuente: (López R. , 2016)

La otra opción sería aplicar average-pooling, el cual trabaja de la misma forma que el anterior, pero en este caso el valor resultante es el promedio de los cuatro pixeles.

Una red neuronal convolucional completa contiene varias combinaciones de capas de convolución y pooling, donde las primeras capas reconocen líneas, mientras las capas más profundas identifican elementos más completos como ojos y rostros, como ya se dijo.



**Figura 1.32** Capas de convolución y pooling en una CNN

**Fuente:** (Calvo, 2017)

Al final de la red neuronal convolucional Figura 1.32, posee una capa completamente conectada, donde cada píxel representa una neurona, de la misma forma que en una red neuronal clásica.

Del análisis realizado a las redes neuronales clásicas y las redes neuronales convolucionales, se concluye que para el procesamiento de imagen es mejor emplear las CNN debido a que reducen el costo computacional al momento del procesamiento de las imágenes, ya que en una CNN se entrena a cada parte de la red para realizar una tarea específica, lo cual reduce el número de capas ocultas y provoca que el entrenamiento de la red sea más rápido, por otra parte, el procesamiento es invariante respecto a la posición en la que se encuentre el elemento buscado dentro de la imagen. Las CNN eliminan la necesidad de una extracción manual de características ya que aprenden directamente, generan magníficos resultados de reconocimiento y se pueden volver a entrenar para nuevas tareas de reconocimiento a partir de redes existentes.

## 1.5. RECONOCIMIENTO DE ROSTROS

El reconocimiento facial es el proceso de identificar o verificar la identidad de una persona mediante su rostro. Captura, analiza y compara patrones basados en los detalles faciales de la persona. El proceso detecta y localiza rostros humanos en imágenes y vídeos, emplea la capturara facial para transformar una información analógica (una cara) en un conjunto de

información digital (datos) basada en los rasgos faciales de la persona. El proceso de coincidencia de caras verifica si dos caras pertenecen a la misma persona.

Paul Viola y Michael Jones en el año 2000, desarrollaron una forma de detectar rostros en forma rápida sin requerir grandes capacidades de hardware, aunque en la actualidad existen soluciones mucho más confiables. A continuación, se analizan dos métodos completos de reconocimientos de rostros, en primer caso se analiza el algoritmo de Viola-Jones y para el segundo caso el método inventado en 2005 llamado Histograma de gradientes orientados (HOG).

### 1.5.1. Reconocimiento de rostros a partir del algoritmo de Viola-Jones

Este método de reconocimiento de imágenes fue desarrollado para realizar la detección de rostros, es uno de los métodos más utilizados en la industria, por su velocidad y su alto porcentaje de acierto. Está compuesto de un conjunto de algoritmos que permiten la detección y el reconocimiento de rostros los cuales se sintetizan en la Figura 1.33.

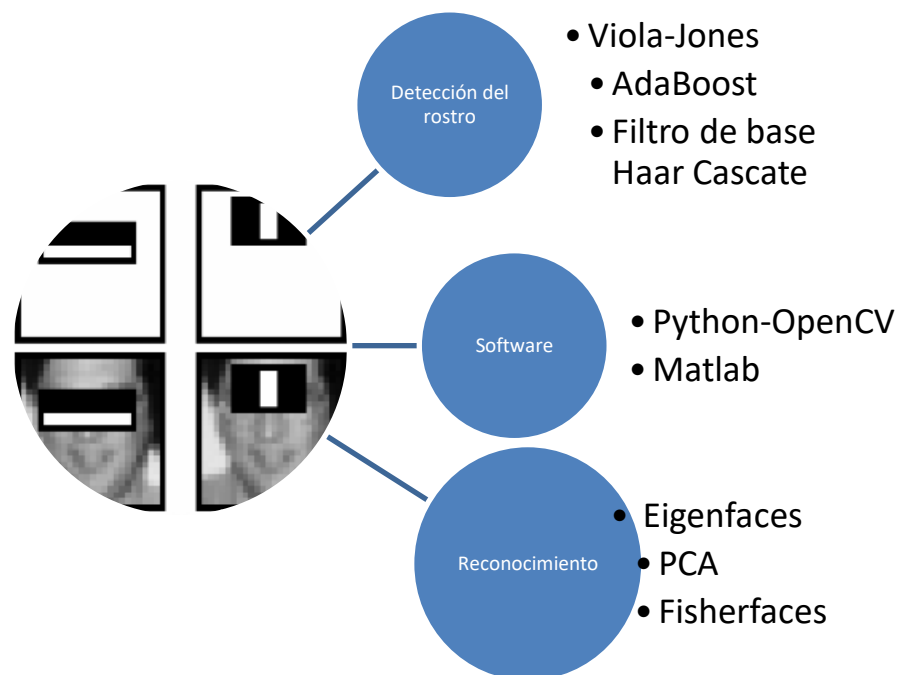


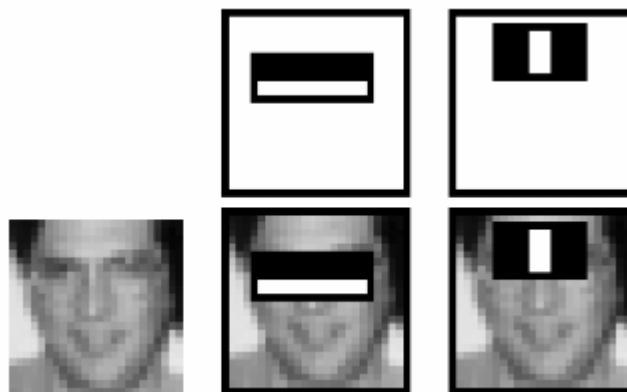
Figura. 1.33. Reconocimiento de rostros a partir del algoritmo de Viola-Jones

Fuente: Autor

- **AdaBoost o Adaptive Boosting.**

Es el algoritmo utilizado por Viola-Jones para realizar el entrenamiento de los clasificadores en cascada que se utilizan en el reconocimiento de rostros en imágenes. Viola-Jones utiliza clasificadores de características para lograr el reconocimiento de los objetos, basándose en características Haar para seleccionar cuál de ellas se ajusta mejor a la definición de un rostro (Papageorgiou, Oren, & Poggio, 1998), se crean clasificadores robustos a partir de clasificadores débiles. En una segunda etapa se compara el rostro detectado con imágenes de muestra mediante el algoritmo de Eigenfaces el cual proporciona una respuesta si existen o no concordancias.

Las características que toma AdaBoost para la identificación rostros permiten un reconocimiento con altos niveles de seguridad. La primera característica analiza la propiedad basada en que la región de los ojos es más oscura que la región de la nariz y las mejillas. La segunda característica se basa en que los ojos son más oscuros que el puente de la nariz en la Figura 1.34, (comienzo de la nariz). (Espinoza & Jorquera, 2015)



**Figura. 1.34 Reconocimiento de los ojos mediante AdaBoost**

Fuente: (Espinoza & Jorquera, 2015)


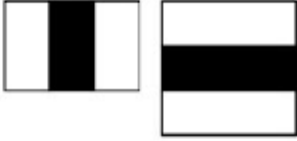
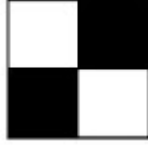
- **Filtro de base Haar Cascade**

Haar Cascade es un algoritmo de detección de objetos de aprendizaje automático *machine learning* utilizado para identificar objetos en una imagen o vídeo y basado en el concepto de características propuesto por Viola-Jones en su trabajo "Rapid Object Detection using a Boosted Cascade of Simple Features" en 2001.

Mediante aprendizaje automático, se entrena una función de cascada a partir de imágenes positivas y negativas, luego el sistema puede detectar objetos en otras imágenes. El algoritmo tiene cuatro etapas: selección de características de Haar, creación de imágenes integrales, formación Adaboost y clasificadores en cascada. Para el análisis de rostros el clasificador necesita una gran cantidad de imágenes con rostros y sin ellos (imágenes negativas) para entrenar al clasificador, mediante la extracción de las características de cada una de ellas.

El primer paso es recopilar las características de Haar, una característica Haar considera regiones rectangulares adyacentes en una ubicación específica en una ventana de detección, resume las intensidades de píxeles en cada región y calcula la diferencia entre estas sumas. En el algoritmo de Viola-Jones se usan tres características como se puede observar en la Tabla 1.2: (Berger, 2018)

**Tabla. 1.1.** Características Haar

Característica de Haar	Definición	Imagen
De dos rectángulos Características de borde	Diferencia entre la suma de los píxeles de ambas regiones rectangulares	
De tres rectángulos Características de línea	Suma de los píxeles de los rectángulos exteriores menos la suma del rectángulo central	
De cuatro rectángulos	Diferencia entre los pares diagonales de los rectángulos.	

**Fuente:** (Berger, 2018)

- **Algoritmo de reconocimiento de rostros Eigenfaces**

Los principales componentes de una imagen de entrada como los ojos, la nariz y las distancias entre estos se denominan Eigenfaces y corresponden a los auto vectores de la

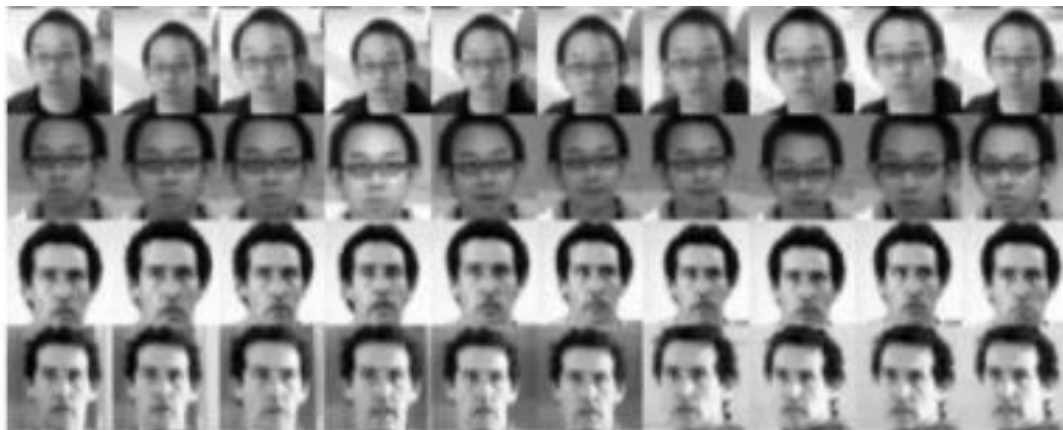


matriz de covarianza de la imagen. El término "eigen" se refiere a un conjunto de vectores propios.

Al emplear una base formada de imágenes "eigen" la dimensión es mucho más pequeña que el conjunto original, este nuevo conjunto se genera por medio de un proceso matemático llamado análisis de componentes principales (PCA) en un conjunto de imágenes con diferentes caras humanas. (Caballero & Reyes, 2017)

El reconocimiento se ejecuta mediante los siguientes pasos:

1. Preparar el conjunto de entrenamiento, las imágenes deben ser tomadas en condiciones de iluminación equivalente, como se muestra en la Figura 1.35, se debe normalizar la imagen, colocar los ojos y la boca en forma horizontal, las imágenes se almacenan en una sola matriz T en que cada columna representa una imagen.



**Figura 1.35** Imágenes de entrenamiento

**Fuente:** (Caballero & Reyes, 2017)

2. Calcular la media del brillo de la imagen y se resta de la imagen original en T.
3. Calcular los auto vectores y los valores propios de la matriz de covarianza S, se obtienen las direcciones en que las imágenes difieren de la imagen media.
4. Se organizan los vectores propios *eigenvectors*, se ordenan los valores propios *eigenvalues* en forma descendente, de esta forma se obtienen los Eigenfaces de la Figura 1.36.



**Figura 1.36** Imágenes Eigenfaces

**Fuente:** (Caballero & Reyes, 2017)

Las imágenes Eigenfaces obtenidas poseen todas las características necesarias para identificar y reconstruir los rostros procesados.

- **Algoritmo Fisherfaces**

Fisherfaces es el método final en el reconocimiento de rostros, realiza un análisis del reflejo de la luz en las expresiones faciales. El algoritmo considera las imágenes de entrenamiento como clases y calcula una matriz de dispersión entre clases y otra dentro de clases. El resultado es una matriz de proyección en la que cada columna es un *Fisherfaces*. El objetivo es determinar cuál imagen del conjunto de entrenamiento es la que más se parece a la imagen de entrada mediante la ecuación de distancia euclidiana que es la fórmula general de distancia entre dos puntos observado en la Figura 1.37.















**Figura 1.37** Imágenes Eigenfaces y Fisherfaces

**Fuente:** (Caballero & Reyes, 2017)

El siguiente es un ejemplo del resultado obtenido al aplicar Fisherfaces para el reconocimiento facial en la Tabla 1.3:

**Tabla 1.2** Resultados de reconocimiento

Imagen de entrenamiento	Imagen de prueba	Resultado
		reconocida correctamente
		reconocida correctamente
		reconocida correctamente
		reconocida correctamente
		reconocida <b>incorrectamente</b>
		reconocida correctamente

**Fuente:** (Anggo & Arapu, 2018)

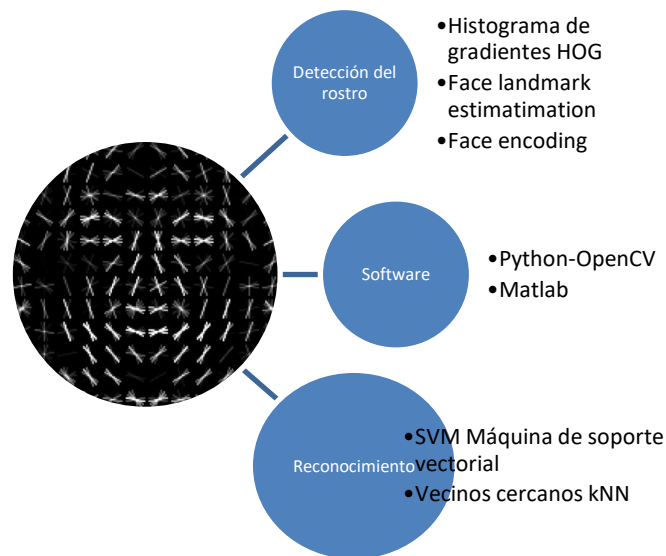
Para finalizar con el análisis de rostros con el algoritmo de Viola-Jones es importante resaltar que este método permite analizar imágenes en tiempo real y su procesamiento es muy rápido, su efectividad es del 93%, es inmune a imágenes introducidas con ruido o borrosidad por otro lado tiene la desventaja de que solo puede reconocer rostros cuando se encuentran de frente, otro factor de falla es que el escalamiento de las imágenes a comparar sean diferentes. (Anggo & Arapu, 2018).

Es sensible a condiciones de iluminación, por lo que puede generar falsos positivos, incluso detectar caras donde no las hay, lo cual lo descarta para utilizar lo a la intemperie, en aplicaciones de seguridad. (Cabrera, 2017)

### 1.5.2. Reconocimiento facial a partir de histograma de gradientes HOG

La detección de rostros a partir de histograma de gradientes HOG utiliza una función “global” para describir un rostro en lugar de una colección de características “locales”. Todo rostro está representado por un único vector de características, a diferencia de muchos vectores de características que representan partes más pequeñas del rostro. (Pardo, 2018)

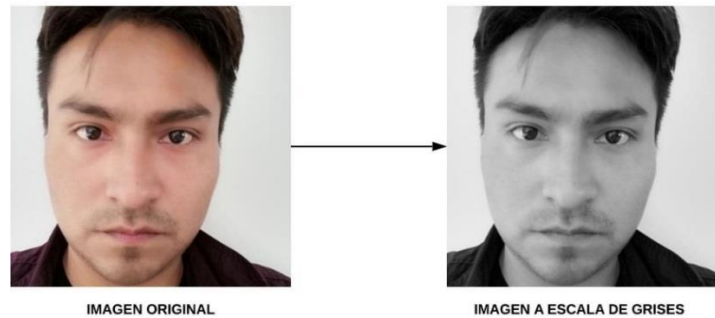
La Figura 1.38 muestra el flujo de trabajo para la implementación de sistema de reconocimiento facial.



**Figura 1.38** Análisis de rostros mediante histograma de gradientes orientados

**Fuente:** Autor

El primer paso en el reconocimiento facial es la detección de rostros. Obviamente, requiere ubicar los rostros en una imagen antes de que se pueda diferenciarlos. Para encontrar caras en una imagen, se convierte la imagen en escala de grises porque no se requiere datos de color.

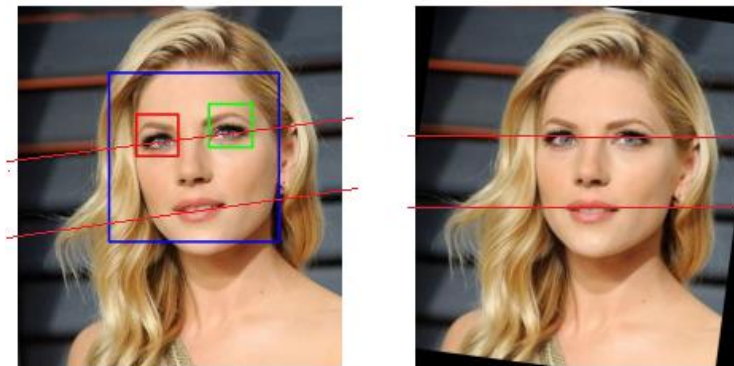


**Figura 1.39** Conversión a) Original b) Escala de grises

**Fuente:** Autor

Parte del proceso de reconocimiento facial consiste en el preprocesamiento de las imágenes para minimizar problemas de iluminación, la conversión de imágenes a escala de grises para un entrenamiento más rápido y la normalización de la posición facial mostrado en la Figura 1.39.

Una vez que los rostros han sido detectados, se debe lidiar con el problema de las diferentes direcciones en las que podrían aparecer. Para resolver esto se deforma cada imagen para que los ojos y los labios estén siempre en el lugar de la muestra en la imagen, es decir, en una posición horizontal. Esto facilita la comparación de caras en los próximos pasos como muestra la Figura 1.40.

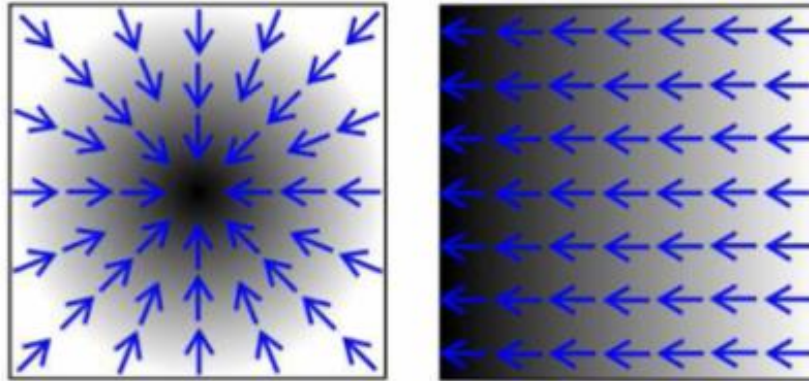


**Figura 1.40** a) Imagen original, b) Rotación de la imagen

**Fuente:** Autor

- **Histograma de Gradientes Orientados HOG**

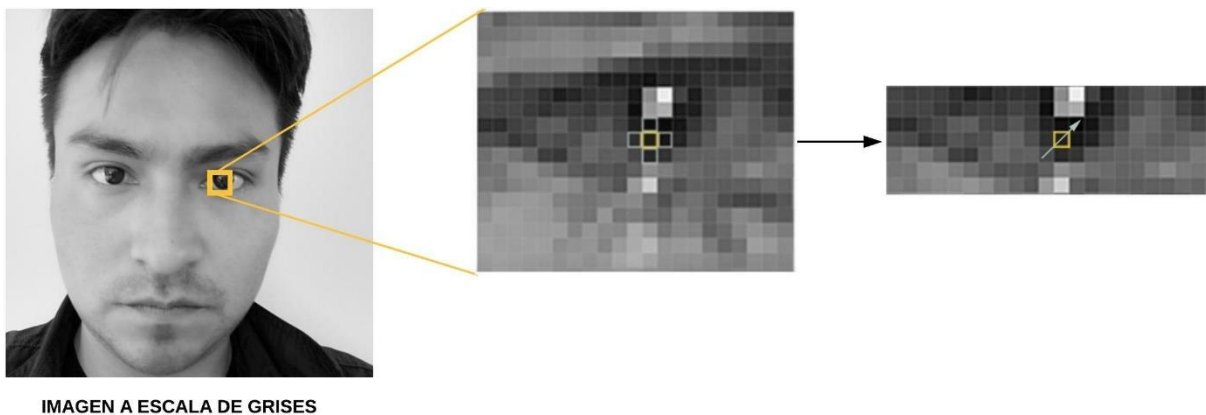
HOG es un descriptor de características, generaliza el objeto en base a referente previo, en este caso de un rostro, que produzca un patrón lo más cercano posible del mismo descriptor de características visto bajo diferentes condiciones. HOG es un método de extracción de características basado en determinar la apariencia y la forma local de un objeto, caracterizado por la dirección e intensidad de la iluminación.



**Figura 1.41 Diagrama de gradientes orientados las flechas se direccionan a las zonas oscuras**

**Fuente:** (Domínguez, 2017)

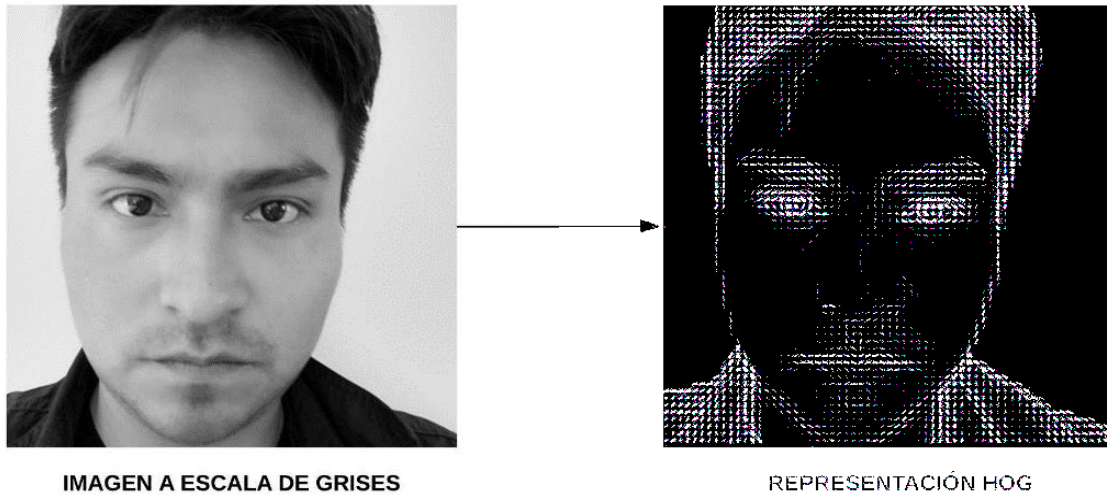
Cada píxel en la imagen es analizado y se lo compara con los píxeles que lo rodean directamente. El objetivo es identificar qué tan oscuro es el píxel actual en comparación con los píxeles vecinos (Domínguez, 2017). Luego se dibuja una flecha que muestra en qué dirección se oscurece la imagen como se muestra en la Figura 1.41.



**Figura 1.42 Procesamiento HOG de la imagen**

**Fuente:** Autor

El proceso se repite para cada píxel en la imagen, hasta que cada píxel sea reemplazado por una flecha denominada gradiente que indica el flujo de claro a oscuro en la imagen. Se puede apreciar en la Figura 1.42.



**Figura 1.43 Diagrama de gradientes aplicado a un rostro, a) Imagen en escala de grises, b) Representación HOG**

**Fuente:** Autor

Esto puede parecer algo aleatorio, pero hay una buena razón para reemplazar los píxeles con gradientes. Cuando se analizan los píxeles directamente, las imágenes realmente oscuras y las imágenes realmente claras de la misma persona tendrán valores de píxeles totalmente diferentes (variación en iluminación). Pero considerando solo la dirección en la que cambia el brillo, tanto las imágenes en realidad son oscuras como las realmente brillantes terminarán con la misma representación exacta como muestra la Figura 1.43. (Catalán & Paternain, 2017)

Para encontrar caras se debe hallar la parte de la imagen que se parezca más a un patrón HOG conocido que se extrajo del conjunto de entrenamiento, como se observa en la (Figura 1.44)

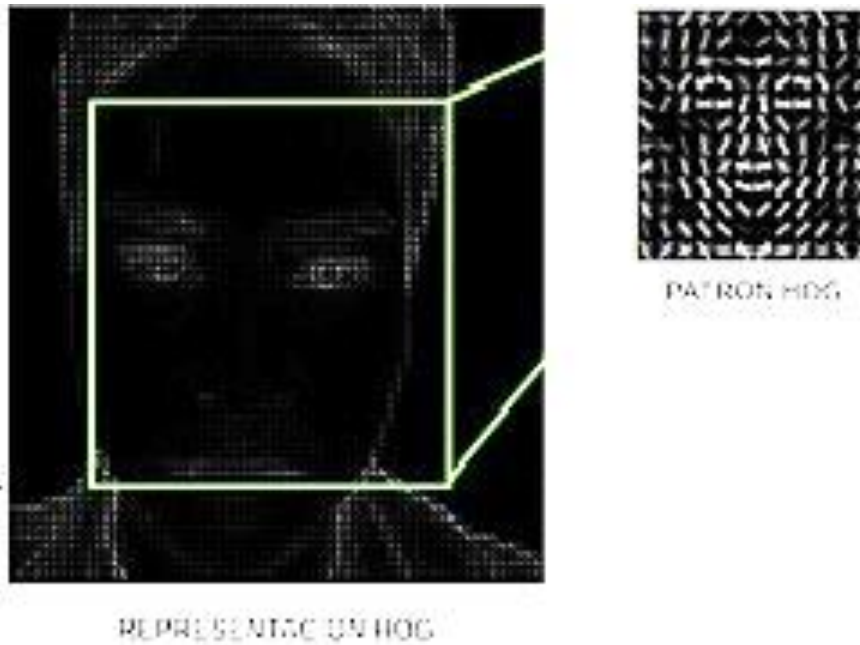


Figura 1.44 Comparación de gradientes, a) Imagen patrón, b) Imagen adquirida

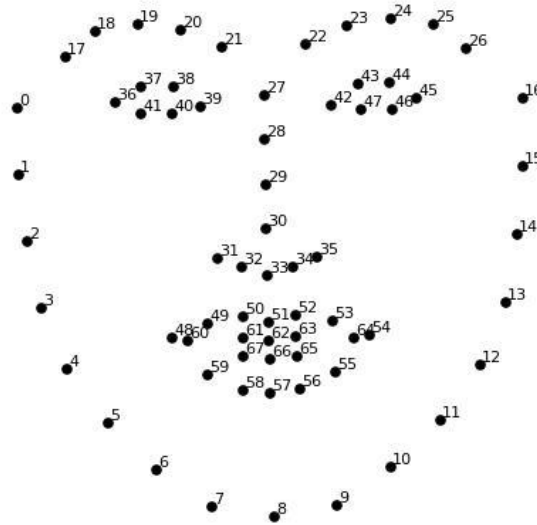
Fuente: Autor

- **Estimación de puntos de referencia facial**

El siguiente paso para el reconocimiento facial se llama estimación de puntos de referencia facial *face landmark estimation*, un enfoque inventado en 2014 por Vahid Kazemi y Josephine Sullivan, establece puntos clave en una cara como la punta de la nariz y el centro de cada ojo. Cada punto de referencia representa un lugar bien conocido en la cara.

La estimación de los puntos de referencia funciona mediante un algoritmo de aprendizaje automático que se entrena para hallar un conjunto conocido de 68 puntos, esos puntos se mueven alrededor del rostro hasta que coincidan con la imagen analizada, se localizan las cejas, los ojos, la nariz, los labios y la línea de la barbilla y debido a que los puntos están predefinidos, se puede saber cuáles puntos forman los ojos, la nariz, y así sucesivamente mostrados en la Figura 1.45.

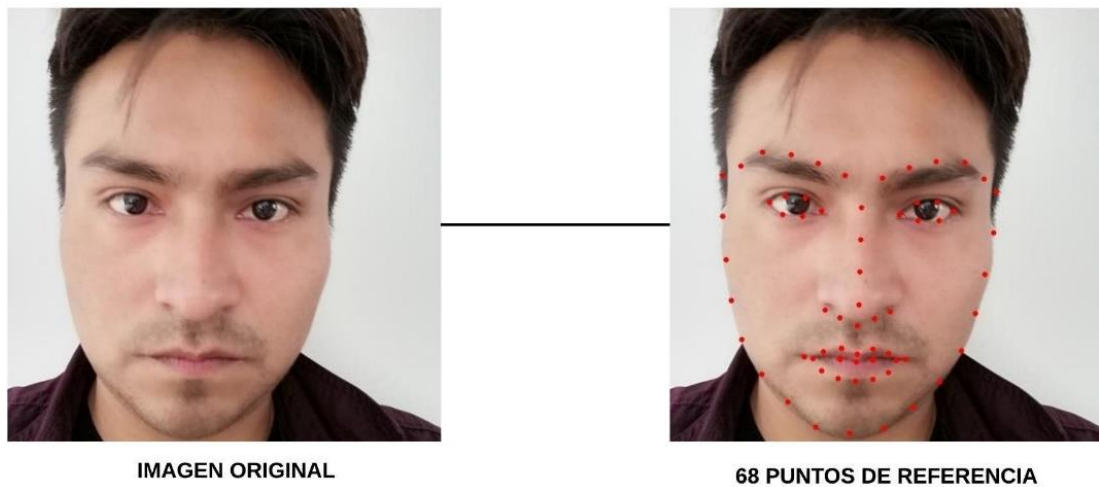




**Figura 1.45 Puntos de referencia facial**

**Fuente:** (Rosembrock, 2017)

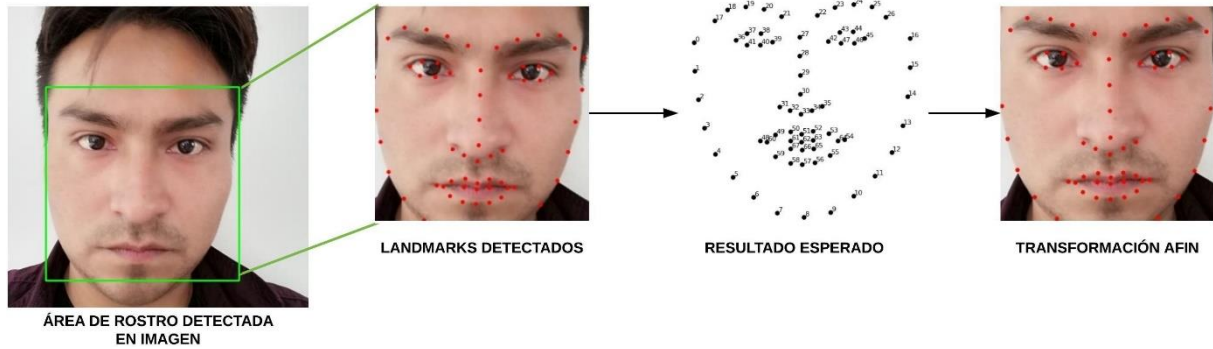
Este es el resultado de ubicar los puntos de referencia de la cara en la imagen de prueba según la Figura 1.46.



**Figura 1.46 Puntos de referencia facial en imagen de prueba**

**Fuente:** Autor

La librería Dlib de Python utiliza la transformación Afín 2D de OpenCV para girar la cara y hacer que la posición de los ojos, la nariz y la boca sean consistentes.



**Figura 1.47** Uso de puntos de referencia facial para normalizar un rostro

**Fuente:** Autor

Las distancias entre los puntos se miden y se comparan con los puntos que se encuentran en una imagen facial promedio. La imagen se gira y se transforma en función de esos puntos para normalizar la cara y se recorta para ser entrada a la red neuronal entrenada. En conclusión, Dlib toma los 68 puntos de referencia de la cara y los usa para calcular la diferencia entre los puntos de referencia de la cara y la ubicación del mismo punto en un modelo de cara promedio de la Figura 1.47.

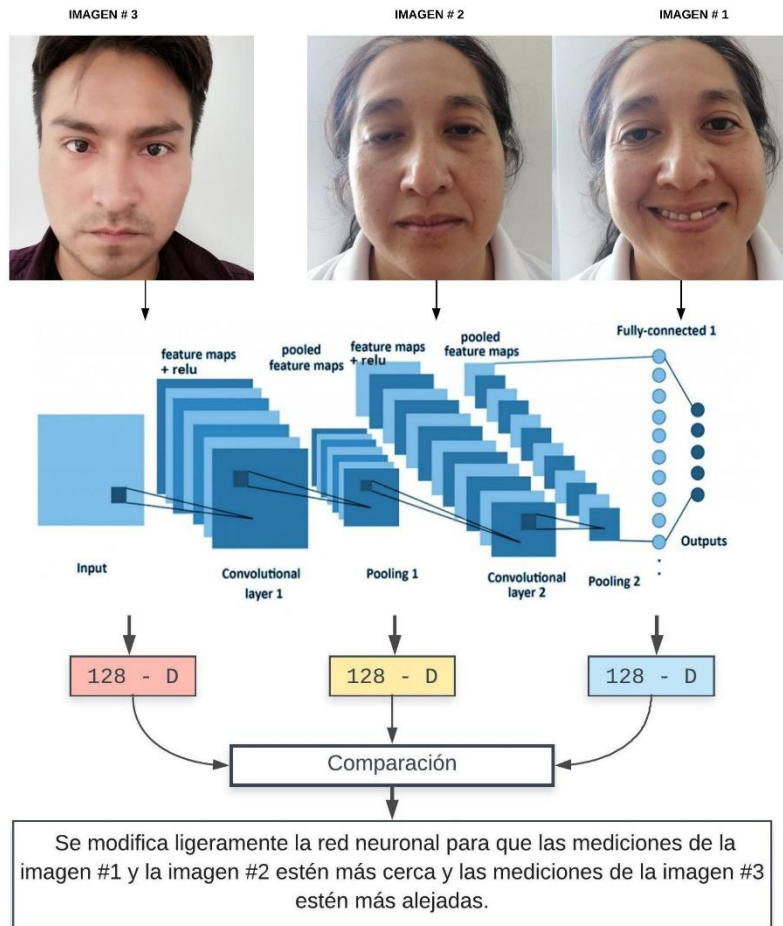
Basándose en esas diferencias de puntos, realiza una transformación en la imagen de la cara para alinear los puntos como, girar el rostro, centrar la imagen y escalarla. La alineación facial es una técnica de normalización, a menudo utilizada para mejorar la precisión de los algoritmos de reconocimiento facial, incluido en los modelos de aprendizaje profundo.

- **Codificación facial - face encodings**

El siguiente proceso consiste en medir las características individuales del rostro para diferenciar unos de otros. Las medidas que parecen obvias a los humanos (como el color de los ojos) realmente no tienen sentido para una computadora que analiza píxeles individuales en una imagen. El aprendizaje profundo *Deep Learning* hace un mejor trabajo que los humanos al determinar qué partes de una cara son importantes para medir.

La solución es entrenar una red neuronal convolucional profunda CNN, para generar 128 mediciones para cada cara. El proceso de entrenamiento funciona al analizar 3 imágenes de rostros a la vez, Cargar un rostro de entrenamiento de una persona conocida, cargar otra foto de la misma persona conocida y cargar una foto de una persona totalmente diferente.

El algoritmo analiza las medidas generadas actualmente para cada una de esas tres imágenes y ajusta ligeramente la red neuronal para asegurarse de que las mediciones que genera para #1 y #2 estén un poco más cerca, mientras que las mediciones para #2 y #3 están un poco más separadas. Se representa en la Figura 1.48.



**Figura 1.48 Entrenamiento de la red neuronal convolucional**

**Fuente:** Autor

Después de repetir este paso millones de veces para millones de imágenes de miles de personas diferentes, la red neuronal aprende a generar de forma confiable 128 mediciones para cada persona. Si se analizara diez imágenes diferentes de la misma persona deberían dar aproximadamente las mismas medidas.

Este proceso de entrenamiento de una red neuronal convolucional requiere una gran cantidad de datos y potencia de computadora. Incluso con una tarjeta de video NVidia Tesla, se requieren aproximadamente 24 horas de entrenamiento continuo para obtener una buena

precisión. Mientras más imágenes analice, el aprendizaje será más profundo y el resultado del reconocimiento será mejor.

Pero una vez que la red ha sido entrenada, puede generar mediciones para cualquier cara, incluso para aquellas que nunca se ha visto, este paso necesita hacerse una sola vez. Para nuevos usuarios es suficiente ingresar las imágenes de las caras en la red pre-entrenada para obtener las 128 mediciones para cada cara. Aquí están las medidas para una imagen de prueba. Como se puede observar en la Figura 1.49.

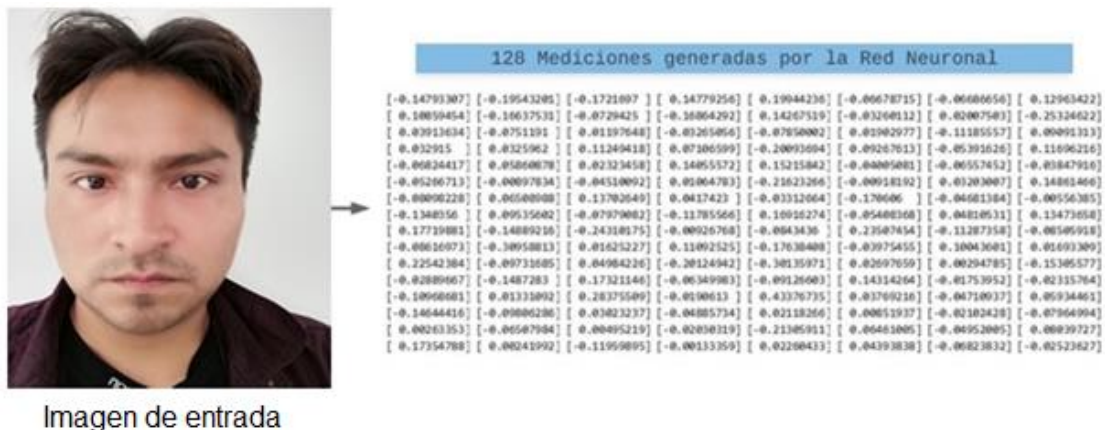


Figura 1.49 Entrenamiento de la red neuronal convolucional

Fuente: Autor

La arquitectura de red para el reconocimiento facial se basa en ResNet-34 mostrado en la Figura 1.50 del documento Deep Residual Learning for Image Recognition de He, Zhang, Ren y Sun. La red en sí fue entrenada por Davis King en un conjunto de datos de 3 millones de imágenes Labeled Faces in the Wild (LFW), demuestra una precisión del 99.38%. Ese modelo pre-entrenado es de dominio público. (He & Zhang, 2016)



Figura 1.50 Base de datos para entrenamiento ResNet-34

Fuente: (Phyton Awesome, 2018)

- **Reconocimiento a partir de clasificadores**

Este último paso consiste en encontrar a la persona en la base de datos conocida, mediante la comparación de las medidas con la imagen de prueba. Puede hacerse mediante cualquier algoritmo básico de clasificación de aprendizaje automático, se presentan a continuación dos clasificadores:

- a) **Support Vector Machines:** Los SVM son modelos de aprendizaje usados para clasificación y regresión. El objetivo es entrenar a un clasificador que pueda tomar las medidas de una nueva imagen de prueba y saber qué persona conocida es la más cercana. Ejecutar este clasificador lleva milisegundos. El resultado del clasificador es el nombre de la persona.

Este método representa en un espacio una serie de clases y trata de encontrar un hiperplano que divida ambas clases de una forma óptima. El cálculo de este hiperplano depende en gran medida de la distribución de las clases y la técnica utilizada para extraer características mostrado en la Figura 1.51. (Domínguez, 2017)

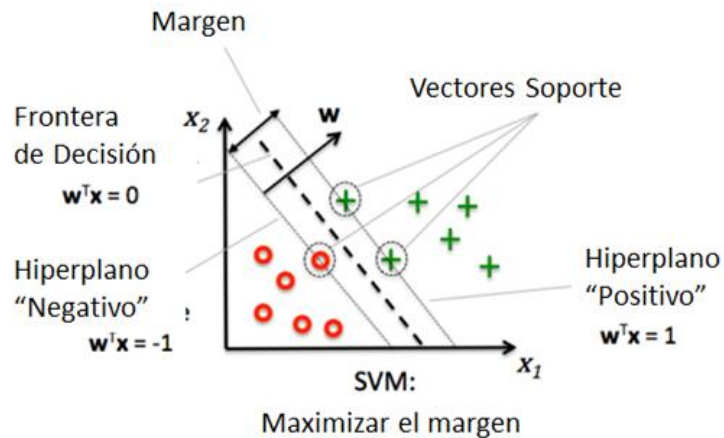


Figura 1.51 Clasificador SVM

Fuente: (Domínguez, 2017)

- b) **Clasificadores K-Nearest Neighbours** El método KNN es una técnica no paramétrica de clasificación de objetos se basa en el análisis de muestras de entrenamiento más cercanas consideradas a partir de un punto de referencia. El algoritmo busca los k vecinos más cercanos al punto y lo clasifica de acuerdo al mayor número de muestras halladas en el área (Morling Global, 2018). En la Figura

1.52., la muestra se clasificaría como triángulo cuando  $k=3$  objetos y como cuadrado, cuando  $k=5$  objetos (línea punteada).

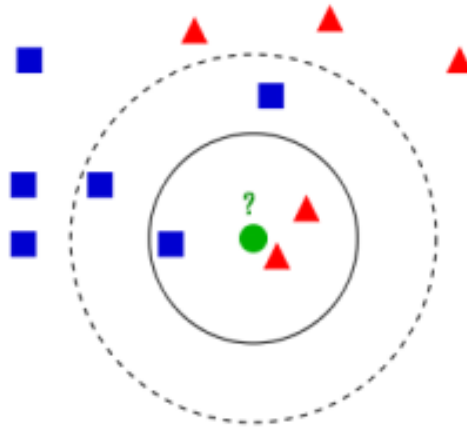


Figura. 1.52. Ejemplo del clasificador K-Nearest Neighbours KNN

Fuente: (Morling Global, 2018)

Para establecer la distancia métrica apropiada el algoritmo utiliza la distancia Mikowski que se define con la siguiente fórmula:

$$d(x^{(i)}, x^{(j)}) = \sqrt[p]{\sum_k |x_k^{(i)} - x_k^{(j)}|^p} \quad (6)$$

Donde:

$p$  valores 1 o 2 correspondiente a la distancia Manhattan y Euclideana respectivamente

$k$  cantidad de puntos vecinos

$x^{(i)} x^{(j)}$  dos puntos conocidos

Como Dalal y Triggs lo descubrieron, el uso de histogramas de gradiente orientados HOG presenta varias ventajas para el reconocimiento de rostros, al emplear gradientes orientados, las imágenes de fondo no afectan en la detección, el nivel de luz sea este bajo o alto no provocan cambios en las características globales del histograma por lo que la detección es estable en cualquier condición de luz. La tasa de reconocimiento en pruebas realizadas demuestra una

efectividad de 97,68% incluso con caras sonrientes (Merchán & Galeano, 2014), lo cual demuestra que el presente método es confiable para la aplicación en sistemas de seguridad.

## **1.6. DETECCIÓN DE OBJETOS**

La detección de objetos es una técnica de visión artificial, se basa en un algoritmo con capacidad de detectar con rapidez varios elementos dentro de una imagen y clasificarlos. Existen diversas aplicaciones como identificación de enfermedades en imágenes, visión robótica, industria, juegos de video y conducción autónoma de vehículos.

Las técnicas de Machine Learning y Deep Learning se han convertido en enfoques generalizados para la detección de objetos y pueden ser elegidos de acuerdo con el problema que se desea resolver. Machine Learning es una técnica efectiva cuando se conoce cuáles son los rasgos de imagen que pueden utilizarse para diferenciarlo de otros objetos. Deep Learning es una técnica que requiere de procesadores potentes y gran cantidad de imágenes etiquetadas para el entrenamiento. (MathWorks, 2019)

La razón de ser de un sistema de reconocimiento de objetos es identificar elementos pertenecientes a una cierta categoría por ejemplo una persona, una mesa o una silla, a partir de una imagen previa. Basado en datos de entrenamiento, el software de reconocimiento ajusta sus parámetros internos para reconocer exitosamente los objetos. (Ruiz & Monroy, 2018)

Existen muchas técnicas para el modelamiento de objetos, basado en sus características, apariencia, modelos geométricos, técnicas de Deep Learning, redes neuronales convolucionales y otras que ya fueron analizadas anteriormente en el reconocimiento de rostros. En este caso se emplea una biblioteca de código abierto para Deep Learning desarrollado por Google denominado tensorflow, una herramienta desarrollada para la detección de imágenes en general.

Tensorflow es extensamente utilizado porque soporta Python, C++, Java y Go. Maneja estructuras de datos como vectores y matrices a los que denomina tensores. Las operaciones matemáticas complejas se realizan en forma cíclica y se las representa en forma de grafos denominados Data flow graph. Utiliza una serie de modelos pre-entrenados listos para realizar predicciones o ser usados en el proceso de transfer-learning: (Castillo, 2018)

- PoseNet: Estimación de poses del cuerpo humano
- BodyPix: Segmentación de partes del cuerpo humano

- Coco SSD: Detección de objetos
- KNN Classifier: Algoritmo KNN de k vecinos cercanos.

Para el funcionamiento de tensorflow es necesario tener instalado Python 3 y las siguientes librerías: Protobuf, Pillow, Lxml, Cython, Jupyter, Matplotlib, Pandas, Opencv-python y Tensorflow.

Para la preparación de tensorflow para la detección de personas son necesarios los siguientes pasos: (Mora, 2018)

1. Preparación del conjunto de datos. Se toma el 80% de imágenes de personas y se las almacena en un directorio “train”. El 20% se almacena en un directorio “test”.
2. Etiquetado del conjunto de datos. Se emplea la herramienta “labelimg” para etiquetar las imágenes antes almacenadas, se generan archivos xml.
3. Generación de registros para la formación. Se crean TFRecords que sirven como entrada de datos de la red neuronal, se requieren los archivos `xml_to_csv.py` y `generate_tfrecord.py` descargables de [github.com](https://github.com).
4. Configuración del entrenamiento. Cree un directorio llamado entrenamiento y dentro de este, un archivo de texto denominado “labelmap.pbtxt” que servirá de mapa para las etiquetas del clasificador.
5. Entrenamiento del modelo. Se requiere el archivo `train.py`, se ejecuta el siguiente comando:

```
python train.py --logtostderr --train_dir=training/ --pipeline_config_path=training/  
faster_rcnn_inception_v2_coco.config
```

6. Exportar gráficos de inferencia. Es necesario crear un gráfico de inferencia para correr el modelo, para crearlo se utiliza la siguiente línea de comando:

```
python export_inference_graph.py --input_type image_tensor --pipeline_config_path  
training/faster_rcnn_inception_v2_coco.config --training_checkpoint_prefix  
training/model.ckpt-XXXX --output_directory inference_graph
```

Se crea el archivo `frozen_inference_graph.pb` que contiene el clasificador de detección de objetos.

Una vez entrenada la red neuronal el sistema se encuentra listo para el reconocimiento de personas. La Figura 1.53 es un ejemplo del reconocimiento de objetos y personas en una imagen:



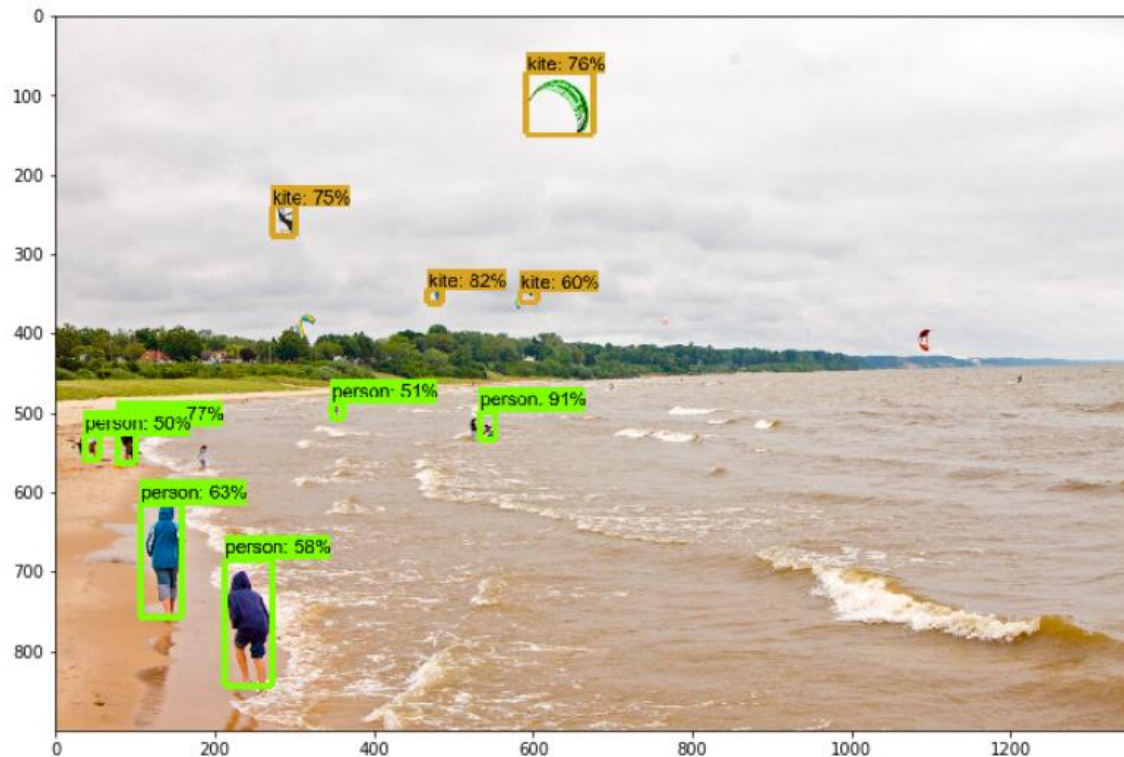


Figura 1.53 Reconocimiento de personas y objetos con tensorflow

Fuente: (Mora, 2018)

## 1.7. LEGISLACIÓN.

Actualmente no existe ninguna normativa legal para el ámbito de la inteligencia artificial en el país, ya que es una ciencia en desarrollo, pero conforme al asesoramiento verbal recibido en diferentes entidades gubernamentales que manejan el ámbito de las Telecomunicaciones y Educativo como el Arcotel, Mintel y el Ministerio de Educación, Conquito, De conformidad a los artículos 103, 173, 174, 178, 190 del Código Orgánico Integral Penal, el presente Proyecto de titulación denominado “Sistema de Video vigilancia mediante visión por computador para el centro de educación inicial 1 del Ministerio de Educación” mencionan, sobre el uso indebido de imágenes y video en:

Pornografía, espionaje, violación a la intimidad y otros que implican las referidas disposiciones legales. Se deslinda al ejecutor del proyecto el señor Leonardo Rubén Arellano con C.C. No 1718816612 de cualquier problema que tenga que ver con el mal manejo de la información y datos almacenados que formen parte de este proyecto a partir de la fecha en que se entregue el proyecto al Ministerio de educación cuya fecha tentativa está dispuesta en el

mes de febrero 2020 ; por cuanto el proyecto se fundamenta en precautelar la integridad de los estudiantes, de conformidad a lo establecido en el Artículo 5 de la Convención Americana sobre los Derechos Humanos que establece: “Derecho a la Integridad Personal.- 1. Toda persona tiene derecho a que se respete su integridad física, psíquica y moral”.

Los lugares donde se instalen las cámaras son de espacio público, no existe un enfoque directo de los menores de edad, tampoco sirve para espionaje; la única cámara que enfoca de manera directa es la que se encontrará a la entrada principal del establecimiento la misma que su enfoque se realiza por encima de 1.50 m de altura, el mismo que es exclusivo para las personas adultas y mayores de edad.

Las cámaras por colocarse a más del señalado en el acceso principal serán en los sitios por donde los delincuentes posiblemente pueden ingresar como son:

El sector que cuenta con cerramientos metálicos que da a la calle principal este se enfocará a la valla metálica y desde el otro punto la otra cámara enfocará al cerramiento que bordea el jardín , de ser necesario y bajo el consentimiento de todos los representantes legales, se procederá a ubicar en lugares específicos en donde se pueda visualizar los rostros de los menores de edad, hay que acotar que ya que el único acceso al establecimiento, es la puerta principal, la cámara de control de acceso mostrará imágenes de la zona considerada como espacio público y en la cual aparecerán más de una persona en conjunto con el menor de edad, ya que lógicamente deberá estar acompañado por un adulto a la hora de retiro o a la hora de ingreso, por lo cual no deberá afectar en lo legal al ejecutor del proyecto, ya que además la cámara de ingreso no enfoca directamente a los menores de edad ni al personal que labora en mencionada institución pública sino directamente a las dos veredas y la calle del acceso principal, la cual es considerada zona de acceso público.

De conformidad con los artículos 229, 230, 232 en lo referente a Delitos contra la Seguridad de los activos de los sistemas de información y comunicación queda a cargo del Ministerio de Educación ya que este deberá implementar los protocolos de seguridad que están bajo su jurisdicción en cuanto a la red de transmisión de información así como también instalar el respectivo antivirus, bajo licenciamiento pagado, con la respectiva empresa de seguridad informática que ellos mantenga con la finalidad de evitar virus, gusanos, spyware, troyanos,

ransomware, adware y otros posibles ataques cibernéticos cuya finalidad es, robar información de cada pc o de dispositivos móviles y de telecomunicaciones.

Se debe realizar también las actualizaciones correspondientes, al sistema operativo y control de estos mecanismos. Para tener una mayor barrera de protección frente a la ciberdelincuencia, de esta manera se deslinda cualquier responsabilidad del creador de mencionado proyecto a partir de la fecha de entrega.

Los datos del personal autorizado que ingresa al establecimiento se almacenarán por un máximo de 7 días en conjunto con la zona video vigilada. Además, se deberá registrar el fichero en una carpeta del sistema operativo Windows de las personas autorizadas donde se guardan las imágenes y se deberá acatar las disposiciones del Ministerio de Educación y la Institución para el archivo de respaldos.

## **CAPÍTULO II**

### **MARCO METODOLÓGICO**

#### **2.1. METODOLOGÍA DE INVESTIGACIÓN, PROCEDIMIENTOS Y HERRAMIENTAS**

##### **2.1.1. Metodología de la investigación**

Para el desarrollo e implementación del proyecto se empleará el método de investigación aplicada, con el fin de resolver problemas de seguridad reales detectados en la institución educativa, en base a un análisis de los requisitos de directivos y usuarios, a fin de mejorar los niveles de seguridad sin interferir en la intimidad de las personas.

El sistema por implementar es la consecuencia práctica de los conocimientos desarrollados en base a la investigación básica realizada por varios autores en el campo de la visión por computadora, reconocimiento facial, aprendizaje de máquina (Machine Learning) y aprendizaje profundo (Deep Learning), quienes han desarrollado algoritmos que permiten la identificación de rostros a través de una cámara de video. Se desea dar un enfoque social para promover la seguridad en el Centro de Educación Inicial No.1 del Ministerio de Educación, para garantizar la seguridad de los niños, personal administrativo y docente, así como a los bienes de la institución. El sistema servirá de apoyo a las actividades de la Empresa de Seguridad del establecimiento.

En cuanto a la profundidad del estudio se utilizará el método de investigación descriptiva ya que se busca medir y clasificar las características faciales de los usuarios para compararlas con una base de datos, no se busca determinar las causas ni las consecuencias de la medición del fenómeno, sino más bien identificar a los individuos en base a técnicas de filtrado de imágenes, extracción de características y algoritmos de reconocimiento facial para tomar decisiones en base a la información obtenida.

Se emplea la investigación bibliográfica para obtener la información requerida, características, configuración e implementación del sistema de reconocimiento facial, las redes de comunicaciones y sistemas de seguridad mediante cámaras de video, además se realizará una

investigación de campo para recopilar información referente a los requerimientos de seguridad del Centro de Educación y los requisitos de los usuarios de los sistemas de seguridad.

### **2.1.2. Procedimientos**

El proyecto actual será implementado en base a investigación bibliográfica del tema en estudio y se verifica en base a técnicas de prueba y error para determinar los algoritmos y dispositivos óptimos, los mismos que se detallan a lo largo de los siguientes capítulos. Algunos de los requisitos, criterios y especificaciones que han orientado la presente implementación, se detallan a continuación:

- a) Observación de campo para determinar los riesgos de seguridad del centro educativo mediante una visita e inspección visual de los accesos e instalaciones, para establecer los puntos débiles en cuanto a seguridad física. Entrevistas personales a las autoridades, personal de seguridad y usuarios para analizar los requisitos de seguridad.
- b) Se analiza posibles soluciones para el control de acceso de las personas y protección de puntos débiles, el cableado y ubicación de las cámaras de vigilancia y del centro de control.
- c) La investigación bibliográfica permite establecer la metodología a utilizar para la implementación del sistema de reconocimiento facial, los requisitos de hardware y software, selección de las cámaras de vigilancia, computadores, redes y algoritmos robustos para el reconocimiento de rostros, frente a todas las posibles variables que puede presentarse en la detección: expresión facial, iluminación, posición, accesorios, maquillaje, calidad de la imagen, etc.
- d) En cuanto a la orientación y distancia de rostros, el algoritmo admite imágenes frontales y con un rango de alejamiento y acercamiento del rostro hacia la cámara.
- e) Se buscará una máxima eficiencia computacional en el procesamiento de la imagen y un mecanismo simple para la alimentación de nuevos usuarios en la base de datos y el entrenamiento del sistema de reconocimiento facial.
- f) El sistema debe ser compatible con otros sistemas de seguridad como el de contraincendios y supeditarse al mismo.
- g) Se determina el marco legal para la implementación del sistema en el Ecuador en base a leyes normativas establecidas por entidades gubernamentales que manejan el ámbito de

las Telecomunicaciones y el ámbito Educativo como el Arcotel, Mintel y el Ministerio de Educación y Conquito.

### **2.1.3. Herramientas**

Las herramientas empleadas para la recolección de la información son las siguientes:

- Se emplea la técnica de entrevistas personales no dirigidas a las autoridades de la institución, personal de seguridad y usuarios del sistema, se emplea como herramienta una hoja de recolección de información y una cámara fotográfica digital. El propósito es obtener información referente a los requerimientos del sistema y expectativas del funcionamiento, identificar puntos débiles de seguridad y obtener fotografías de los usuarios para la base de datos del sistema.
- Se utiliza la técnica de observación de campo, como herramienta una hoja de recolección de información y google maps, para obtener datos acerca de riesgos de seguridad en las instalaciones y establecer el mejor lugar para la instalación de cámaras de seguridad, el centro de control y monitoreo del sistema.

## CAPÍTULO III

### PROPUESTA

La solución que se desea implementar busca cumplir con los requerimientos de los usuarios en cuanto a seguridad mediante un sistema de video vigilancia que integra componentes de cámaras de video vigilancia y de visión por computador para el control de ingreso del personal autorizado y la detección de personas como medio de seguridad física.

El diseño está dirigido a mejorar los niveles de seguridad en el establecimiento educativo y se han tomado en consideración que cumpla con los siguientes requisitos: (Aceves, 2013)

- Flexibilidad, debe ser concebido para soportar crecimiento futuro.
- Modularidad, ya que debe permitir modificar o agregar tecnología sin que se comprometa la funcionalidad integral.
- Multidisciplinario, permite integrar tecnología electrónica, control, redes de comunicación, informática, para tener resultados correctos.
- Seguridad, debe permitir replicarse tanto local como remotamente para garantizar el acceso a la información desde puntos de respaldo del sistema.
- Compatibilidad, ya que debe permitir la conexión y transferencia de la información.
- Intuitivo, a medida que se integran las aplicaciones de inteligencia se vuelve sencillo de usar y brinda soporte para la toma de decisiones.
- Escalabilidad, ya que el sistema puede conectarse a n cámaras conforme unas variaciones en el software diseñado a principio.

El sistema empleará tecnología de punta con algoritmos de inteligencia artificial para el reconocimiento de rostros, una base de datos de acceso, detección de personas que ingresan o están en el interior del establecimiento, control de puerta de acceso incluso en caso de fallos de energía comercial y el monitoreo permanente de las zonas críticas mediante cámaras de seguridad de alta definición con detección de personas.

Hoy en día acorde a las innovaciones tecnológicas es posible aplicar la inteligencia artificial, especialmente la parte de la visión por computador aplica como módulo a los sistemas

de video vigilancia. En este caso específico se lo desarrolla para el control de acceso al establecimiento que trata de erradicar actos delincuenciales junto a un sistema electrónico mixto acoplado mediante protocolos de comunicación y el cual generará un registro de imágenes tomadas en tiempo real de las personas que ingresan al establecimiento y acercan su rostro a la cámara y en conjunto con la detección de personas que dará una alerta auditiva al personal de seguridad de turno.

Las personas serán previamente registradas mediante la adquisición de fotografías con técnicas de Machine Learning y redes neuronales para de esta manera el sistema permita el ingreso a las personas autorizadas.

Desde el interior de la escuela se ha visto también la necesidad de instalar cámaras que realicen el proceso de detección de personas el cual se podrá visualizar bajo demanda mediante el software respectivo, para que se incluya en el monitoreo del área.

Los sistemas básicos de este proyecto de titulación son los siguientes:

- Reconocimiento facial mediante una cámara de video y apertura automática de la puerta de acceso al establecimiento solo con acercar el rostro de la persona autorizada y registrada en la base de datos.
- Detección de personas en movimiento mediante visualización de imágenes captadas por la cámara de video vigilancia.

### **3.1. RECONOCIMIENTO FACIAL**

El sistema de reconocimiento facial se instalará en la garita ubicada en la puerta de acceso a la institución. Se propone una solución mediante un sistema electrónico mixto. El primero, el control de apertura de la puerta, que consta de sensor de proximidad, sensor magnético, placa Arduino, pantalla LCD 5" y una chapa magnética. Estos estarán conectados al módulo de reconocimiento facial conformado por un computador que ejecuta el programa que identifica plenamente a una persona en base a sus rasgos característicos y técnicas de Machine Learning. Además, envía señales de control para la apertura de la puerta, el sistema realiza las siguientes acciones:



- Reconocimiento facial
- Autorizar o denegar el acceso a la institución.
- Almacenar información relativa al usuario autorizado que ingresa, fecha y hora en la cual ha solicitado el acceso mediante el sistema de visión por computador.
- Generar una alarma audible ante cualquier intento de ingreso no autorizado.
- Informar si la puerta está abierta.
- Proveer una alarma audible ante cualquier intento de ingreso no autorizado.
- Facilitar un control completo de la apertura de la puerta de acceso desde los puestos de operador autorizado y en caso de emergencias.
- Mostrar en una pantalla al usuario su ubicación para un mejor encuadre durante el proceso de reconocimiento facial.

La electrónica que controla el funcionamiento de la cerradura electromagnética va asociada a los sensores y dispositivos mediante un microcontrolador, relé y demás dispositivos de audio para que en conjunto con el sistema de reconocimiento facial permita el acceso al personal autorizado. El cableado de alimentación deberá tomarse del punto eléctrico más cercano que facilite el funcionamiento.

Los sistemas de reconocimiento de rostros y detección de personas se encuentran interconectados entre sí a través de una intranet. Los componentes de la intranet son cámaras IP, router, repetidores de señal, computador que realice en proceso de reconocimiento de rostros, computador que realice el proceso de detección de personas y computador que muestre en tiempo real el monitoreo de los subprocesos de visión por computador tratados en este proyecto.

La Figura 3.1 presenta un diagrama de bloques del sistema de reconocimiento facial, detección de personas y apertura de la puerta.

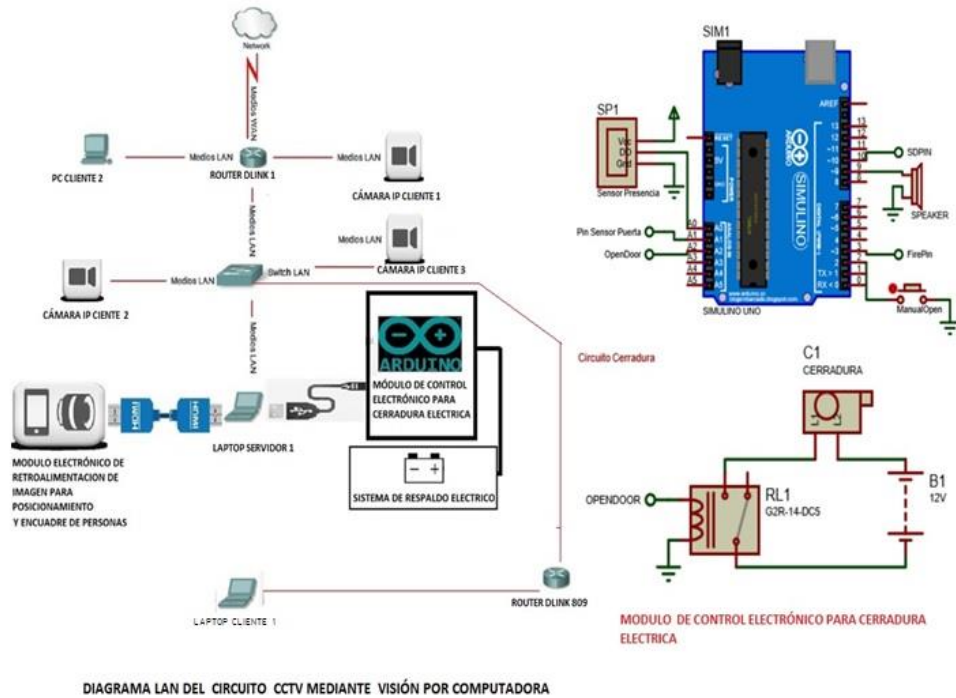
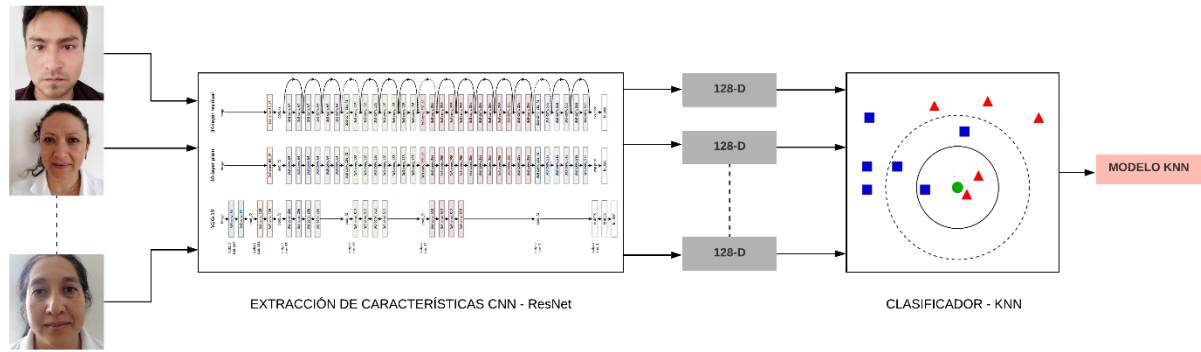


Figura 3.1 Diagrama de bloques general del sistema de CCTV mediante visión por computador

Fuente: Autor

Para la identificación facial se seleccionó la técnica de reconocimiento facial a partir de histograma de gradientes HOG, el cual utiliza una función “global” para describir un rostro, a través de un único vector de características. Este sistema contiene dos algoritmos independientes: Algoritmo de entrenamiento del conjunto de datos de usuarios y algoritmos de reconocimiento de rostros.

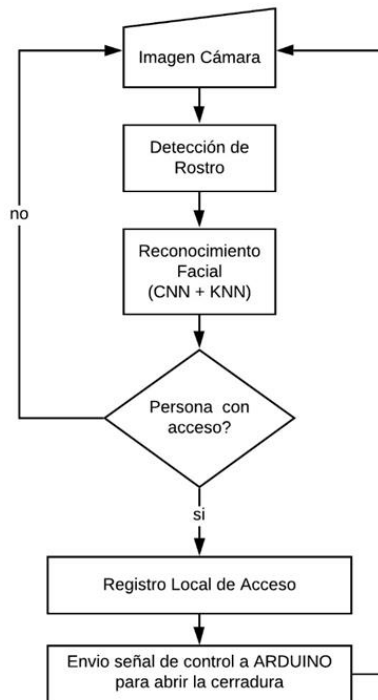
El algoritmo de entrenamiento del conjunto de datos de usuarios permite el entrenamiento de la red neuronal con el rostro de los usuarios autorizados. Para el entrenamiento se requieren por lo menos cinco fotografías de cada usuario, siendo 20 fotografías una muestra ideal para garantizar la calidad de la información ingresada y el procesamiento sea más preciso, las fotografías del rostro deben ser de frente, tener una resolución de 500 x 500 pixeles, con fondo blanco y deben ser tomadas con un solo tipo de cámaras durante todo el tiempo de uso del sistema. A continuación, se muestra el diagrama de flujo de trabajo para el entrenamiento del conjunto de datos de usuario como se observa en la Figura 3.2.



**Figura 3.2** Flujograma de entrenamiento de conjunto de datos

**Fuente:** Autor

La Figura 3.3 muestra el algoritmo de reconocimiento de rostros y realiza el procesamiento de la información proveniente de la cámara de video vigilancia, además del proceso de reconocimiento en base a histogramas de HOG, redes neuronales convolucionales y clasificadores KNN, el siguiente flujograma presenta un detalle de los pasos requeridos para la identificación.



**Figura 3.3** Flujograma de reconocimiento de rostros

**Fuente:** Autor

La tarjeta Arduino se encargará del control de la cerradura magnética de la puerta y detectar cuando la puerta se encuentre abierta, la señal de control la recibe desde el computador de reconocimiento facial vía USB (SRF). El diagrama de flujo del algoritmo de control de la tarjeta Arduino como se muestra en la Figura 3.4.

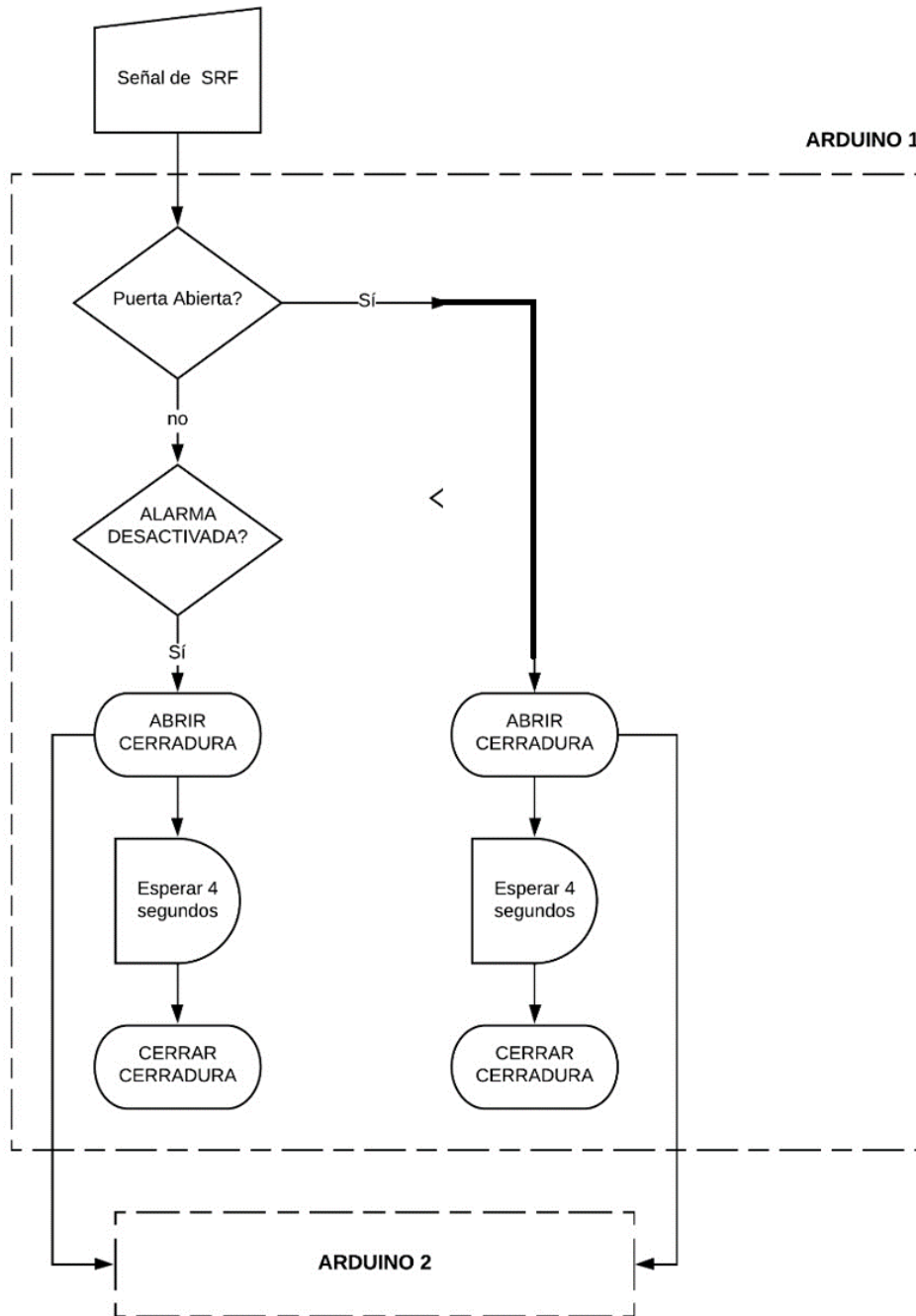


Figura 3.4 Flujograma de control mediante arduino

Fuente: Autor

### 3.2. DETECCIÓN DE PERSONAS

Este sistema se instalará en dos cámaras IP de alta resolución que serán ubicadas en el interior de la institución, la primera dirigida hacia la puerta de ingreso y la segunda dirigida hacia el patio de juegos.

Las cámaras fijas una vez instaladas y configuradas, quedan enfocadas a una zona de visión concreta. Se instalarán fijadas a la fachada mediante brazos o elementos de fijación adaptados. La elección de la óptica de las cámaras permitirá observar incidentes y detectar a las personas que discurran por las zonas que se desea monitorizar.

Las cámaras se deberán adaptar a las condiciones climáticas, de temperatura y de luminosidad del lugar de instalación. Dispondrán de iluminación mediante infrarrojos que permitan la captación de imágenes tanto de día como de noche.

El sistema está compuesto por dos cámaras IP, computadores con el software de detección de personas mediante TensorFlow, este software busca patrones similares en las imágenes para detectar las siluetas humanas. El flujograma explica el proceso de detección de personas en la Figura 3.5.

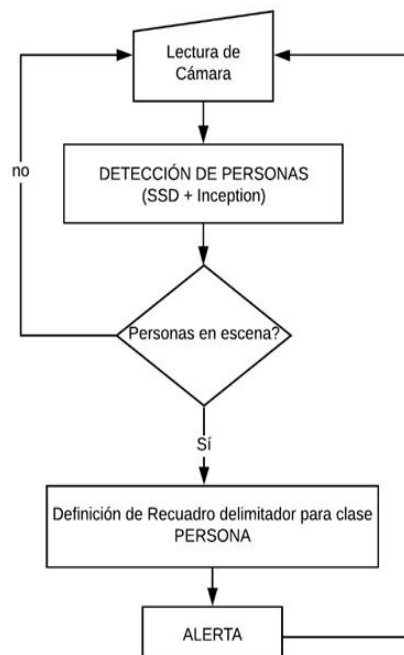


Figura 3.5 Flujograma de reconocimiento de personas

Fuente: Autor

### 3.3. ESPACIO PARA GRABACIÓN DE IMÁGENES Y VIDEO

Las cámaras digitales de tecnología IP ofrecen imágenes de alta resolución y grabación en tiempo real, el software de reconocimiento facial desarrollado con técnicas de Deep learning posibilita capturar imágenes y video en tiempo real por ende da más protección al establecimiento al ser utilizado en procesos de investigación. Para ahorrar espacio en disco se utilizan técnicas como el empleo de almacenamiento de pocos frames por segundo y la activación de la grabación a través de un software de CCTV tradicional vinculado dentro del algoritmo de reconocimiento facial al CCTV mediante visión por computadora mediante un botón, la cual puede realizarse en una zona determinada o por franjas horarias, lo que se conoce como generación de eventos.

Con el uso del estándar H.264, se consigue elevadas tasas de compresión para evitar altos consumos de ancho de banda y espacio de almacenamiento en disco duro de la laptop, con lo cual se sustituye el uso de NVR (Network Video Recorder).

La gestión del conjunto de cámaras IP y módulo de reconocimiento facial instalados puede realizarse de manera centralizada desde un único punto y gracias al protocolo IP, puede hacerse incluso en modo remoto conforme se planifique con el departamento de tecnologías del Ministerio de Educación y se logre indexar al área que se encarga de monitorear los diferentes puntos de la mencionada entidad.

El sistema de video vigilancia es evolutivo. El aumento del número de cámaras no implica una modificación mayor en la arquitectura del sistema; arquitectura que a su vez permite infinidad de opciones de visualización e integración.

Para realizar el cálculo de almacenamiento y ancho de banda del sistema de grabación y reproducción es necesario definir la cantidad total de cámaras a implementar, las cuales son 3 fijas, la resolución de grabación que para las cámaras será de 2Mp. Adicional se define el Frame Rate (imagen por segundo), para las cámaras que se van a utilizar conforme lo define la *Tabla 1.4*.

Se definen los porcentajes de movimiento de objetos basado en la ocupación de las áreas, es decir la cantidad de objetos que en determinada área son constantes para la vista de la cámara y estos son:

50% Static

40% Standart

10% Busy

Otra consideración es el tiempo de grabado en 24 Horas, el tipo fija de exteriores, cabe indicar que el tiempo de grabado del diseño es de 7 días. Con esta información ya descrita se procede a utilizar el software Disk Calculator de la compañía ProyTelecom para calcular la necesidad del ancho de banda, que arrojó los resultados mostrados en la Figura 3.6.



Figura 3.6 Cálculo de espacio en disco duro para grabación de imágenes y video

Fuente: Autor

## 3.2. HERRAMIENTAS DE SOFTWARE

### 3.2.1. Lenguaje de programación Python

El sistema de reconocimiento facial se desarrolla en Python, se ha elegido para este proyecto por sus altas prestaciones, es un lenguaje de programación de propósito general, simple, versátil y de rápido desarrollo. Es un lenguaje interpretado que no requiere de un compilador para ser ejecutado. Python se ha hecho muy popular por las siguientes razones: (Alvarez, 2003)

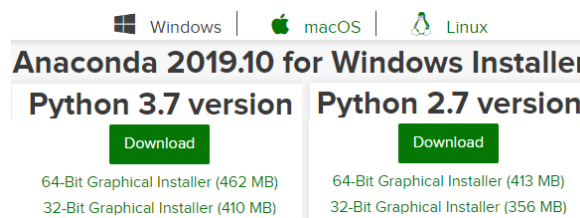
- La cantidad de librerías que contiene, tipos de datos y funciones incorporadas en el propio lenguaje, que ayudan a realizar muchas tareas habituales sin necesidad de tener que programarlas desde cero.
- La sencillez y velocidad con la que se crean los programas. Un programa en Python puede tener de 3 a 5 líneas de código menos que su equivalente en Java o C.
- La cantidad de plataformas en las que se puede desarrollar, como Unix, Windows, OS/2, Mac, Amiga y otros.
- Python es gratuito, incluso para propósitos empresariales.
- Es ordenado y limpio, es muy legible y sus módulos están bien ordenados.
- Dispone de algoritmos y librerías de Machine Learning y redes neuronales.
- Extenso soporte a través de foros e internet.

Para la creación de un ambiente de programación machine learning con Python es necesario seguir la siguiente agenda:

1. Descargar e instalar Anaconda
2. Creación del entorno virtual Face\_Recognition
3. Instalar el IDE Spyder
4. Instalación de paquetes requeridos

## 1. Descargar Anaconda

Anaconda mostrada en la Figura 3.7 es una distribución de Python, que instala Python y un conjunto de paquetes útiles para el análisis científico. Brinda una gran cantidad de funcionalidades que permiten desarrollar aplicaciones en forma más eficiente, rápida y sencilla. Es libre y de código abierto, con gran cantidad de documentación y una gran comunidad.



**Figura 3.7 Instalación de Python y Anaconda**

**Fuente:** (Anaconda, 2019)



## 2. Creación del entorno virtual Face\_Recognition

En la Figura 3.8 se muestra un entorno virtual es un espacio independiente a la instalación local, con el objetivo de aislar los recursos y librerías. Gracias a este concepto se puede tener distintos entornos virtuales con diferentes versiones de Python o de una librería concreta. Si se tiene versiones menores q la 3.3 o 2.8 es necesario descargar un fichero denominado get-pip.py de la pagina oficial de Python. Las razones por las que deberías usar entornos virtuales son:

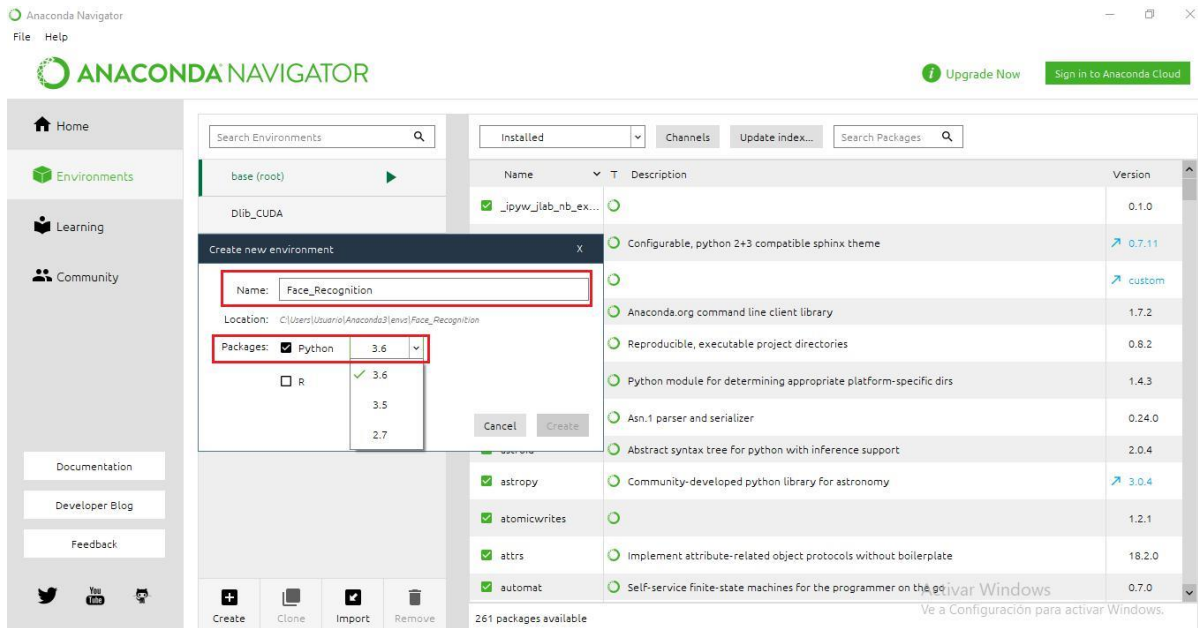
**Organización:** es conveniente tener controladas las librerías y las versiones con las que trabaja para que no existan problemas de incompatibilidades, y para ello, es mejor organizar todo en entornos virtuales, así, no tener la instalación de Python con multitud de librerías instaladas.

**Estabilidad:** es un hecho que las librerías cambian de versión muy a menudo, esto puede ocasionar que los programas ya no funcionen.

Cuando se tenga un código extenso desarrollado, llega una versión nueva de una librería, para esto gestiona los entornos virtuales de manera que permite trabajar con versiones distintas de repositorios o ficheros.

**Trabajar con diferentes versiones de Python:** existen muchas aplicaciones en la versión 2.7 de Python y algunas no cuentan con soporte para Python 3. Se crea distintos entornos virtuales se puede cambiar de versión de Python en dos líneas de código.

Inicialmente se ejecuta la interfaz gráfica de usuario (GUI) Anaconda Navegador. En la sección Environments seleccionar Create:



**Figura. 3.8 Creación del entorno virtual Fase\_Recognition**

**Fuente:** Autor

Definir el nombre del nuevo entorno virtual y la versión de Python a utilizar. En la ventana del lado derecho Descriptions, se muestran las librerías instaladas en el entorno virtual con sus versiones.

### 3. Instalar el IDE Spyder

Spyder mostrado en la Figura 3.9, es un potente entorno de desarrollo interactivo para el lenguaje Python. Posee funciones avanzadas de edición, pruebas interactivas, depuración e introspección y un entorno informático numérico. Gracias al soporte de IPython (intérprete interactivo mejorado de Python) y bibliotecas populares de Python como NumPy, SciPy o matplotlib (trazado interactivo 2D / 3D).

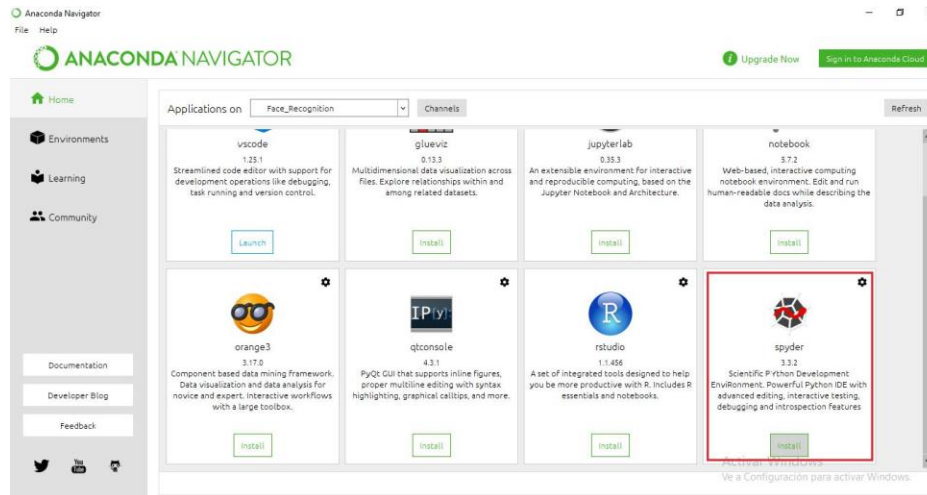


Figura. 3.9 IDE Spyder

Fuente: Autor

#### 4. Configuración de paquetes requeridos

La configuración de paquetes de Python se realiza a través de la ventana de comandos de Windows en el entorno virtual Face\_recognition creado en el paso anterior. Para el desarrollo de la aplicación de reconocimiento facial bajo Python, OpenCV y técnicas de Deep Learning, se requiere la instalación de las siguientes librerías externas:

**NUMPY** es una extensión de Python, que le agrega mayor soporte para vectores y matrices, constituye una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices. Versión 1.15.4.

**OPENCV** (Open Source Computer Vision) es una librería software open-source de visión artificial y Machine Learning. OpenCV provee una infraestructura para aplicaciones de visión artificial. Version 3.2.0.

**DLIB** es un moderno kit de herramientas de C++ que contiene algoritmos de aprendizaje de máquina (Machine Learning) y herramientas para crear software complejo en C ++. Es una librería “para resolver problemas del mundo real”. Se utiliza en la industria y en el mundo académico en una amplia gama de dominios, incluye la robótica, dispositivos integrados, teléfonos móviles y grandes entornos de computación de alto rendimiento. Las licencias de código abierto de Dlib permiten usarlo en cualquier aplicación, de forma gratuita. Versión 19.9.0

**PILLOW** es una librería gratuita que permite la edición de imágenes directamente desde Python. Soporta una variedad de formatos, incluidos los más utilizados como GIF, JPEG y PNG. Es una bifurcación de PIL Python Imaging Library (PIL) para versiones 3.x de Python. Versión 5.3.0.

**Face\_recognition.** Biblioteca para el reconocimiento facial basada en técnicas de Deep learning que permite definir el vector de características para cada rostro. Basado en dlib, un moderno kit de herramientas de C++ que contiene varios algoritmos de aprendizaje automático que ayudan a escribir aplicaciones sofisticadas basadas en C ++. Versión 1.2.3

**Imutils** es un paquete de funciones elementales de procesamiento de imágenes básicas, tales como la traslación, rotación, cambio de tamaño y visualización de imágenes. Versión 0.5.1.

Con la configuración de las librerías anteriores, el software Python se encuentra listo para el desarrollo de aplicaciones de reconocimiento facial. Como ya se anotó el lenguaje de programación elegido es de distribución gratuita, así como todas sus librerías, es un lenguaje moderno que facilita la programación, detección de errores y dispone de una amplia documentación y comunidad en línea para el desarrollo de aplicaciones y resolución de errores. Estas son las razones por las que fue elegido como software de programación en el presente proyecto.

### **3.3. ANÁLISIS DE COSTOS Y TIEMPO**

El análisis de costos permite establecer la viabilidad económica del proyecto, a la vez que sirve de base para posibles mejoras y optimización. En el Anexo 1 se presenta el listado de materiales empleados con sus valores respectivos.

Todos los implementos mencionados en Tabla 1.4 tendrán un costo de 3906,63 dólares americanos, los cuales serán asumidos por el señor Leonardo Arellano quien es el desarrollador de este proyecto de graduación.

### **3.4. VENTAJAS DEL PRODUCTO**

El sistema de video vigilancia mediante visión por computador para el Centro de Educación Inicial N°1 del Ministerio de Educación presenta varias ventajas para la institución y sus usuarios, de las cuales se mencionan las siguientes:

- El sistema de video vigilancia basado en visión por computadora presenta un avance tecnológico en seguridad, ya que este permite realizar procesamiento de imágenes en tiempo real, identificar rostros, reconocer personas y activar sistemas en base a eventos como es el caso de la chapa magnética. Lo cual no se puede realizar con un sistema de video vigilancia tradicional.
- Es un sistema que no necesita de energía eléctrica para su funcionamiento y operación.
- El uso de computadoras con procesador Intel i7 en el procesamiento de video e imágenes para la detección y reconocimiento permite un procesamiento rápido, ya que este tipo de procesamiento en tiempo real requiere gran cantidad de recursos computacionales.
- Las cámaras de alta definición IP permitirán realizar un alto nivel de reconocimiento de las imágenes. Mientras que el método de almacenamiento y compresión de imágenes ahorra el espacio en disco duro requerido.
- El cableado y sistema se encuentra muy bien reguardado y protegido ya que para evadirlo de alguna manera la infraestructura física deberá ser derrumbada.
- El método de reconocimiento de rostros a partir de histogramas de gradientes HOG garantiza un alto nivel de seguridad en la identificación de los usuarios registrados con diferentes niveles de iluminación, imágenes de fondo y accesorios. Por otra parte, este método requiere menor cantidad de recursos computacionales que el método de Viola-Jones.
- Es un sistema que no solo cuenta con un respaldo sino varios respaldos de información.
- La red interna entre los sistemas de video vigilancia permite compartir información y realizar el monitoreo desde varios sitios a la vez, incluso de ser necesario, se puede conectar a redes externas.
- El costo-beneficio de la implementación del sistema actual supera las expectativas del proyecto y es mucho menor que el costo comercial de un sistema de este tipo.

## CAPÍTULO IV IMPLEMENTACIÓN

### 4.1. DESARROLLO

Las tareas por realizar dentro de la instalación del sistema de video vigilancia mediante visión artificial serán las siguientes.

#### 4.1.1. Desarrollo del software de reconocimiento de rostros.

El software de reconocimiento de rostros fue desarrollado mediante el método de reconocimiento facial a partir de histograma de gradientes HOG para lo cual se desarrollaron dos programas. El primer programa consiste en el algoritmo de entrenamiento del conjunto de datos de usuarios, se utiliza cada vez que se desea agregar o eliminar usuarios de la base de datos del sistema. Para agregar usuarios se requieren por lo menos cinco fotografías digitales para entrenar el sistema. Como se muestra en la Figura 4.1.

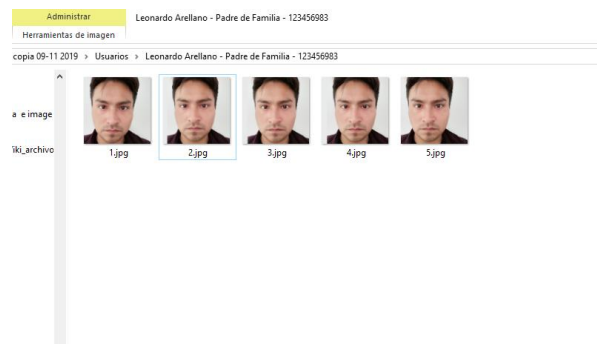


Figura 4.1 Número mínimo de fotografías para entrenamiento

Fuente: Autor

- **Algoritmo de entrenamiento del conjunto de datos**

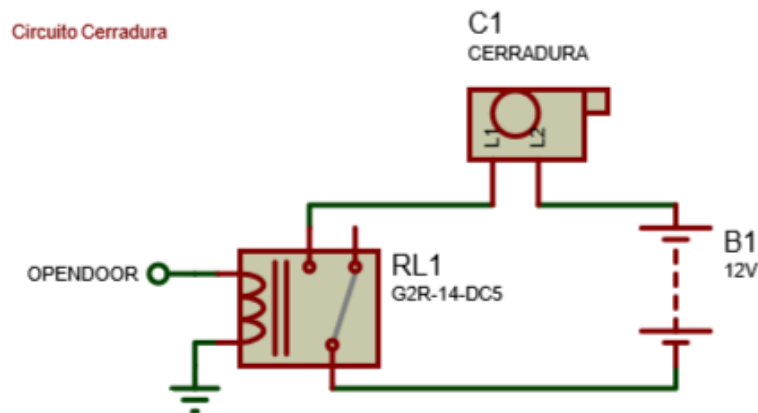
El archivo Train.py se presenta en el Anexo 2, contiene el algoritmo KNN de entrenamiento del conjunto de datos de usuarios, basado en el análisis de vecinos cercanos explicado anteriormente. El programa se explica mediante comentarios en el mismo.

- **Algoritmo de reconocimiento de rostros**

El segundo algoritmo que se presenta en el Anexo 3, es el programa principal y se encarga del reconocimiento de rostros, este programa es ejecutado permanentemente para la lectura de las personas que se acercan a la cámara, realiza el reconocimiento facial, registra a la persona y envía una señal de control a la cerradura magnética en caso de haber una identificación positiva. Se han incluido comentarios explicativos a lo largo del programa.

#### 4.1.2. Circuito de control de la puerta de acceso.

En la Figura 4.2 se puede observar el diagrama para la puerta de acceso que es controlada por una cerradura electromagnética de 12Vcd de alimentación, controlada por la tarjeta Arduino a través de un relé.



**Figura. 4.1** Control eléctrico de cerradura de la puerta de acceso

**Fuente:** Autor

La tarjeta controladora Arduino, realiza el enlace del software con el mundo real. Recibe señales de control a través del puerto USB y en base a estas controla la apertura de la cerradura magnética, detecta si la puerta está abierta o cerrada, recibe señales de un sensor de presencia para informar de la presencia de personas al frente de la cámara de reconocimiento facial, tiene dos pines para la apertura de la puerta, en forma manual y a través del sistema contra incendios del establecimiento. Además, emite información de audio a los usuarios, la misma se encuentra almacenada en una memoria SD de la Figura 4.3.

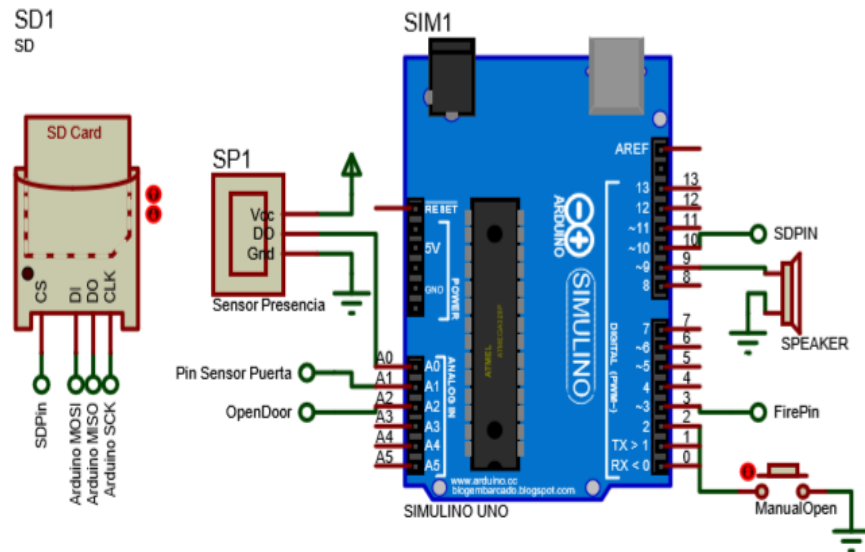


Figura 4.2 Circuito electrónico de control de la puerta de acceso

Fuente: Autor

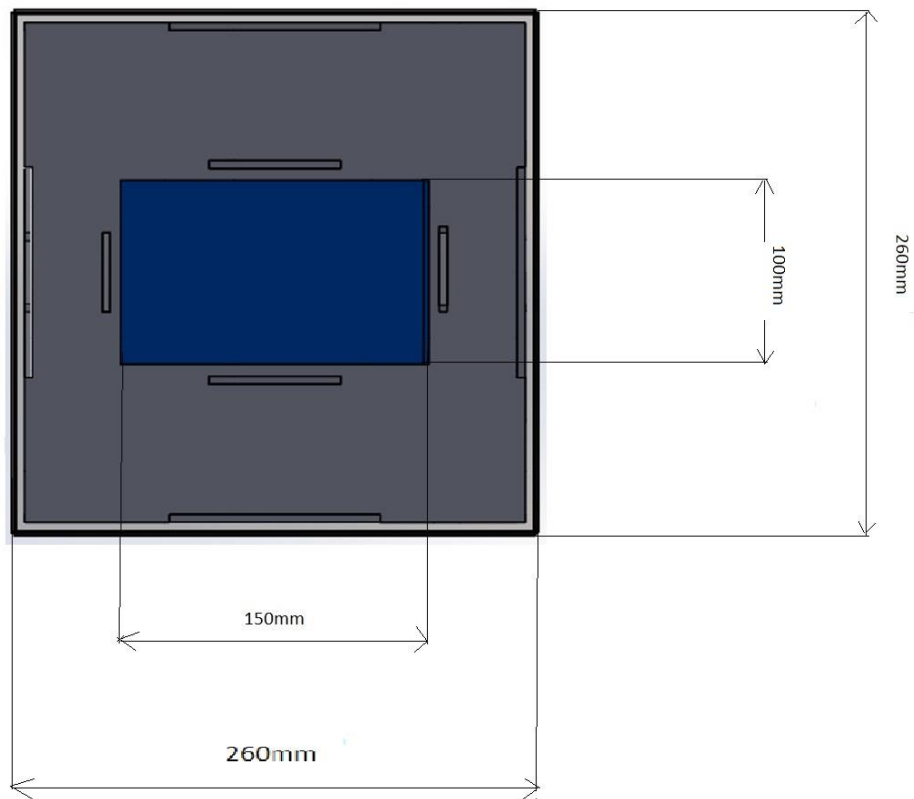
#### 4.1.3. Desarrollo del software de control de la tarjeta Arduino.

El Anexo 4 presenta el software desarrollado en el IDE de Arduino para el control del hardware explicado en el punto anterior. Se han agregado comentarios a lo largo del programa para su mejor comprensión.

#### 4.1.4. Diseño, construcción y ensamble de cajas.

El material a utilizarse para el modulo de retroalimentación de imagen para usuarios, mostrado en la Figura 4.4, es en acrílico, por ser un material mas resistente que los utilizados en impresiones 3D, el mismo que cuenta con lugares específicos donde deben ser empotrados todos los elementos electrónicos como el sensor, la pantalla, entre otros.





**Figura 4.3 Módulo electrónico de retroalimentación de imagen para ubicación y posicionamiento de usuario**

**Fuente: Autor**

#### **4.1.5. Desarrollo del software de reconocimiento de personas.**

El reconocimiento de personas basado en Tensorflow utiliza una base de datos de objetos pre-entrenada, por lo que no es necesario realizar el entrenamiento. Los objetos que puede reconocer se encuentran en el archivo cocolabels.py, que se presenta en el Anexo 5.

#### **4.1.6. Instalación de cámaras, dispositivos y red.**

Instalar las cámaras IP de exterior en las ubicaciones propuestas con sus anclajes.

Tender el cableado desde el Router correspondiente hasta cada cámara. Se deberán incluir todos los elementos necesarios tales como extensores de Ethernet o las canalizaciones que puedan resultar necesarias.

Conexionar y configurar las cámaras, enfocándolas hacia las zonas que se desea cubrir.

Instalar la PC que funcionará como visualizador y procesador de los archivos de visión por computador en la ubicación acordada con las autoridades, incluyendo la electrónica de red necesaria para su acceso a la red local.

Instalar la laptop que servirá de visualizador en el puesto del vigilante, debidamente conectado a la red para que pueda acceder a visualizar los procesos de reconocimiento facial y detección de personas de las cámaras, así mismo como al sistema de video vigilancia

Las tareas para realizar dentro de la instalación del sistema de control de accesos serán por lo tanto las siguientes. Estos trabajos deberán contar siempre con la aprobación del director del establecimiento o documentos habilitantes que respalden la ejecución del proyecto de tesis.

Instalar la placa electrónica en un lugar cercano a la puerta principal, con todo el sistema de reconocimiento facial, tender el cableado necesario desde el cuarto de control y la alimentación eléctrica correspondiente a la electrónica del sistema de seguridad hasta los respectivos puntos Se deberán incluir todos los elementos necesarios tales como extensión de cable o las canalizaciones que puedan resultar necesarias.

Conexionar y configurar los microcontroladores, así como también calibrar y ubicar los sensores.

Instalar el software de reconocimiento facial en la laptop que debe estar en la garita principal que permitirá tener una mejor visualización al guardia de turno del área en el cual brinda sus servicios.

Instalar el software de detección de personas que se conectará a través de la red LAN al pc de monitoreo mediante visión por computador.

Realizar las pruebas y puesta en marcha del sistema para su correcto funcionamiento.

## **4.2. SEÑALIZACIÓN**

El centro contará con letreros similares a los que se muestra en la (Figura 4.5) que indiquen que posee cámaras de seguridad y dar aviso de las personas que acceden al

establecimiento, en este caso se dará una advertencia del uso de la inteligencia artificial en la operación de las cámaras.



Figura 4.4 Letrero de Información

Fuente: Autor

#### 4.2.1. Puesta en marcha del ingreso de Usuarios a la base de datos del sistema.

La interfaz mostrada en la Figura 4.6 permite el ingreso de nuevos usuarios a la base de datos para la apertura automática y registro de sus imágenes mediante reconocimiento facial en un espacio de memoria dentro de la PC.

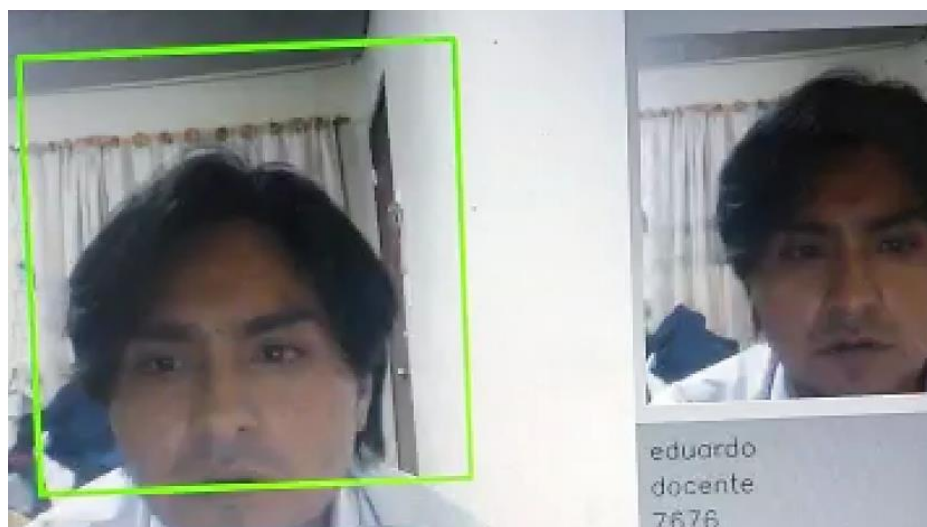


Figura 4.5 Ingreso de nuevos usuarios

Fuente: Autor

#### 4.2.2. Puesta en marcha del sistema de respaldo energético para el control de la puerta de acceso

En la Figura 4.7 se muestra el sistema de respaldo energético el cual es un mecanismo que se tomó en cuenta ante el corte de energía eléctrica que por cualquier motivo se dé, dentro de la instalación educativa, además de ser un mecanismo que funciona para eventos naturales como, incendios, terremotos, entre otros.



Figura. 4.6 Respaldo Energético

Fuente: Autor

#### 4.2.3. Puesta en marcha del sistema de detección de personas

Cuando la cámara entra en funcionamiento y se pone en funcionamiento el software de detección de personas se obtiene un proceso similar al mostrado en la Figura 4.8.

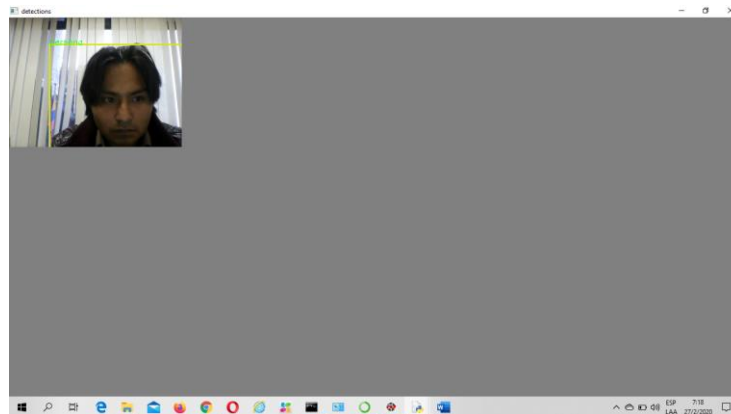


Figura 4.7 Proceso de detección de personas.

Fuente: Autor

### 4.3. PRESUPUESTO

En el estudio de ingeniería para determinar el coste de ejecución del proyecto se tuvo como objetivo primordial el elaborar un presupuesto referencial de un sistema CCTV mediante visión por computador para el centro de educación inicial del Ministerio de Educación se analizaron los trabajos que se deben ejecutar para realizar el proyecto, así como la descomposición de los costos, la mano de obra directa, incluida su maquinaria, en la ejecución del proyecto.

**Tabla 1.4. Presupuesto**

MODELO	UD	DESCRIPCIÓN	CANTIDAD	PRECIO	TOTAL
MATERIALES					
	Unidades	LAPTOP TOSHIBA	1	1200	1200
	Unidades	LENOVO i7 5ta generación.	1	1400	1400
WIC20	Unidades	CAMARA IP EPCOM CUBICA	1	131.06	131.06
AE-CP315	Unidades	COOLING PAD	1	12.050	12.050
C270HD	Unidades	WEB CAM LOGITECH	1	80.51	80.51
56262	Unidades	CERRADURA RIM LOCK SILK	1	37.5	37.5
	Unidades	CONTROLADOR DE CARGA SOLAR	1	15.89	15.89
NAXOS 500	Unidades	SWITCH FAST ETHERNET	1	11.57	11.57
AE-HUB	Unidades	HUB USB 2.0 DE 4 PUERTOS	1	4.86	4.86
	Unidades	CARGADOR ADAPTADOR 12 V 2A	3	4.28	4.28
		ARDUINO UNO	1	5.0	5.0

	Unidades	MODULO RELE DE UN CANAL	1	2.32	2.32
		MÓDULO MP3	1	5.0	5.0
50Nk	Unidades	SENSOR INFRAROJO	1	15.18	15.18
	Unidades	SENSOR MAGNÉTICO	1	2.15	2.15
DIR 809	Unidades	ROTEADOR WI-FI ROUTER DLINK AC 750 DUAL BAND	1	36.5	36.5
	Unidades	LCD DISPLAY 5 PULGADAS	1	66.96	66.96
	Unidades	BATERIA SECA RECARGABLE DE 12 V	1	23.21	23.21
T12HD	Paquetes	AMARRAS 30 CM	1	4.43	4.43
MANO DE OBRA					
	Horas	OFICIAL	30	4	120
	Horas	AYUDANTE	30	2	60
MEDIOS AUXILIARES					
		SEÑAL ÉTICA DE SEGURIDAD INDUSTRIAL	1	2.5	2.5
	Metros	EXTENSIÓN DE CABLE USB	15	1.74	26.10
	Metros	EXTENSIÓN DE CABLE HDMI	3	1.41	4.24
	Metros	EXTENSIÓN USB PARA ARDUINO	1	5.36	5.36
	Caja	CABLE UTP CATEGORÍA 5E 100M	1	43.33	43.33
	Rollo	CABLE AWG 14	1	45.54	45.54

	Metros	CANALETAS 24/14MM ADHESIVA	20	1.52	27.93
	Metros	CANALETAS 48/28 MM ADHESIVA	2	5.6	11.20
	Metros	CABLE INSTALACIÓN GPT 18 AWG	10	0.21	2.10
	Unidades	CAJETÍN DE PASO	2	17.0	34
	Unidades	TOMAS SOBREPUESTOS	2	3.31	6.62
	Metros cuadrados	CORTE Y GRABADO ACRILICO DE 3 mm	(1/8) m <sup>2</sup>	22	22
	Unidades	CONECTOR DB15	1	3.5	3.5
	Unidades	TACOS F8	16	0.04	0.64
	Unidades	TACO FISHER F6	20	0.02	0.40
	Unidades	TORNILLOS 6*3/4	20	0.02	0.40
	Unidades	TORNILLOS VARIOS	25	0.02	0.50
	Unidades	BORNERA PIN INDUSTRIAL	1	1.7	1.7
	Paquetes	CUCHILLAS	1	1.29	1.29
	Unidades	CRIMPADORA DE IMPACTO	1	3.57	3.57
	Unidades	JACK RJ45 CAT 6	1	3.48	3.48
	Unidades	UNION HDMI HEMBRA	1	3.20	3.20
SUBTOTAL					3488.07
IVA					418.56
TOTAL					3906.63

Fuente: Autor

## 4.4. PRUEBAS DE FUNCIONAMIENTO

Se realizó las pruebas, configuraciones y puesta en marcha del sistema para su funcionamiento como se muestra en las siguientes Figuras 4.8; 4.9; 4.10; 4.11.

### 4.3.1. Pruebas de funcionamiento del sistema de control de acceso.

El modulo de retroalimentación conformado por la cámara Ip como dispositivo de adquisición de imágenes, un sensor de presencia y una pantalla TFT, permiten que el usuario se encuadre dentro de los límites que el sistema necesita para su funcionamiento óptimo como se observa en la Figura 4.9



**Figura 4.8** Módulo de retroalimentación para ubicación de rostro e ingreso.

**Fuente:** Autor

### 4.3.2. Pruebas de funcionamiento del sistema de reconocimiento de personas

La imagen corresponde al cliente de reconocimiento facial en donde se visualiza una interfaz con asignación de identidad de usuarios de la base de datos previamente construida. Para el caso actual se observa que la persona es un usuario que no consta en la base de datos por lo que el sistema procede a no reconocerlo y por lo tanto no permite la apertura automática de la puerta como se muestra la Figura 4.10.





**Figura 4.9** Sistema reconociendo a personas no autorizadas.

**Fuente:** Autor

### 4.3.3. Pruebas de funcionamiento del sistema de detección de personas

Para el cliente detección de personas, cuando el sistema valida la presencia de individuos en la escena se torna la imagen transmitida al servidor de color roja evidenciando que hay alguien en el área como se muestra en la Figura 4.11.



**Figura 4.10.** Detección de personas

**Fuente:** Autor

#### 4.3.4. Interfaz de funcionamiento del sistema cctv mediante visión por computadora

En el sistema CCTV mediante visión por computador, al ejecutar el primero el servidor arranca el proceso que llevará a visualizar una interfaz grafica como se muestra en la Figura 4.12 en donde se observará las imágenes recibidas por los clientes de detección de personas y de reconocimiento facial



**Figura 4.11** Interfaz para visualización del guardia de turno.

**Fuente:** Autor

## 4.5. ANÁLISIS DE RESULTADOS

- Para validar el funcionamiento del sistema CCTV mediante Visión por computador, se toma en cuenta realizarlo en diferentes horarios durante el día y la noche: Como eje principal se obtuvo los resultados esperados con un nivel de confiabilidad del 97%.

Durante el proceso se realizó la validación de almacenamiento y la creación de carpetas individuales por fechas, en las circunstancias bajo elementos como gafas gorras y situaciones nocturnas con lo que se obtuvo las capturas correspondientes de cada imagen con un intervalo de 4 segundos por cada toma debido a un tiempo de encuadre que el usuario debe experimentar al momento de colocarse en frente de la cámara.

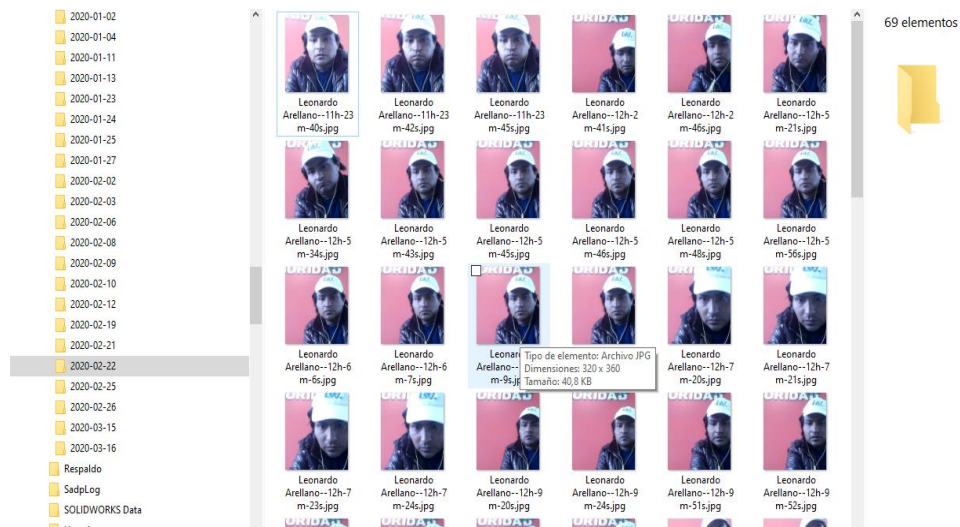
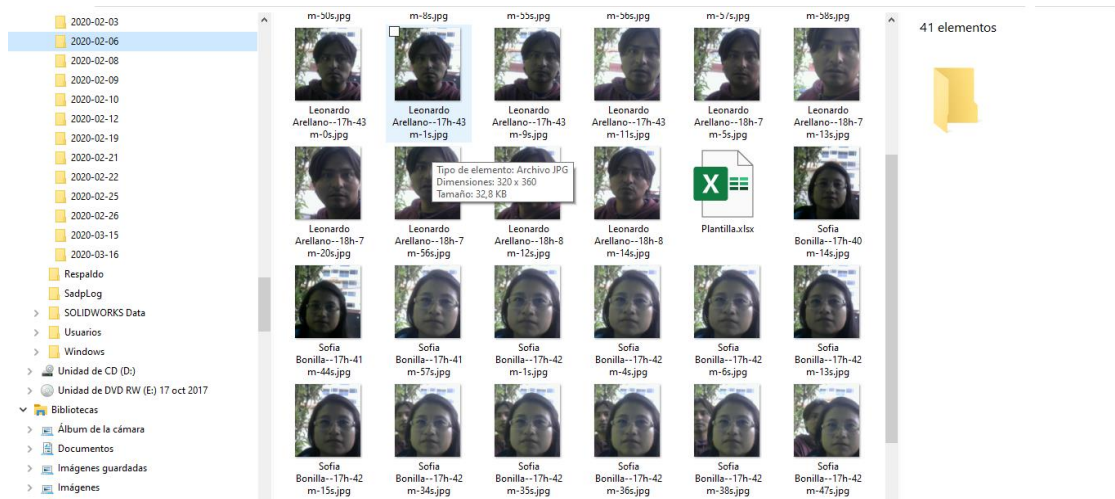


Figura 4.12 Personal registrado y autorizado utilizando accesorios para la cabeza

Fuente: Autor

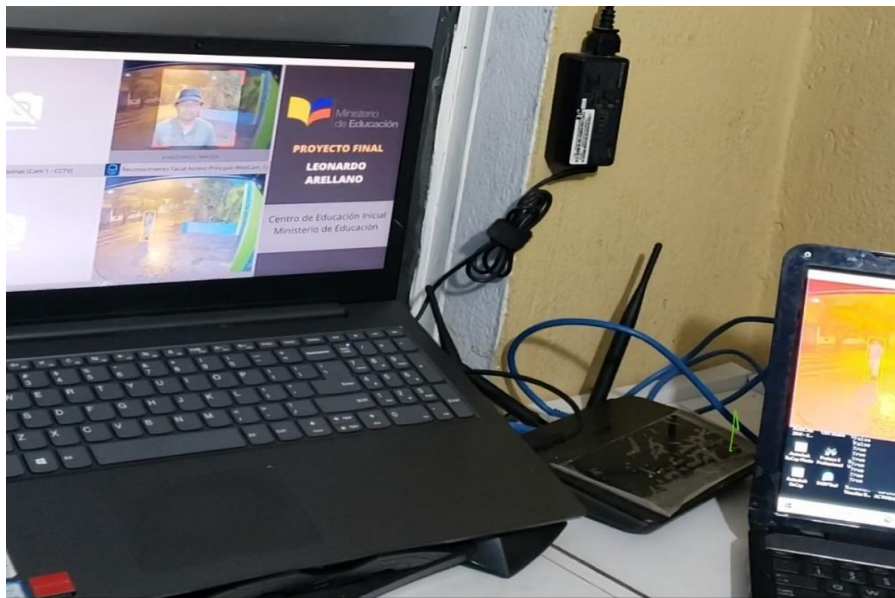
El sistema cliente de reconocimiento facial es capaz de capturar personas con gafas o diferentes accesorios de vestimenta. Como se observa en las Figuras 4.13; 4.14.



**Figura 4.13 Personal identificado y almacenado en la base de datos con lentes.**

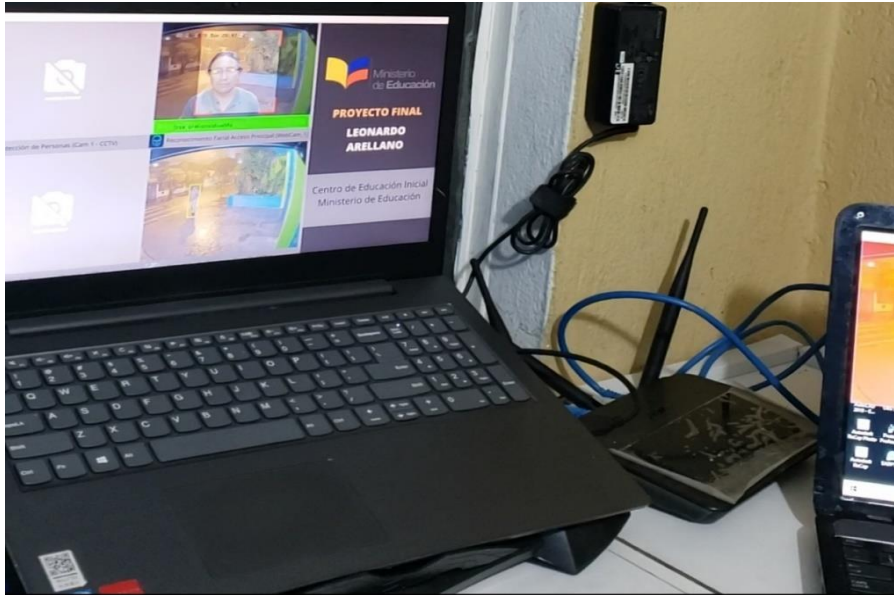
**Fuente: Autor**

- Ante problemas de iluminación nocturna se procedió a utilizar una cámara Ip con tecnología infrarroja para que el sistema esté operativo las 24 horas. Como se observa en las Figuras 4.15 ; 4.16.



**Figura 4.14 Proceso de identificación y detección de personas parte 1**

**Fuente: Autor**



**Figura 4.15** Proceso de identificación y detección de personas pate 2.

**Fuente:** Autor

- Se procedió a instalar una clave de acceso para apertura de la puerta bajo responsabilidad del guardia de Turno mediante el teclado de la Figura 4.17.



**Figura 4.16.** Teclado para ingreso de clave para apertura de puerta

**Fuente:** Autor

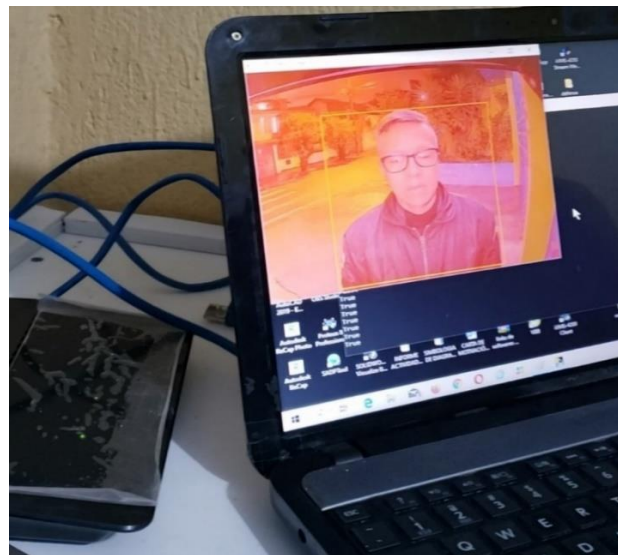
- Ante problemas de iluminación como las que genera el Sol, se procedió a variar el WRD de la cámara cuya tecnología funciona como un filtro para una mejor calidad de adquisición de imágenes la cual se observa en un día soleado mediante la Figura 4.18.

- El sistema de detección de persona emite alertas audibles y visibles para el día y la noche para una mejor respuesta del personal de seguridad de turno. (Figura 4.19)



**Figura 4.17 Validación del sistema en horario diurno.**

**Fuente: Autor**



**Figura 4.18 Detección de personas horario nocturno**

**Fuente: Autor**

## CONCLUSIONES

Una vez finalizada la implementación del sistema de video vigilancia mediante visión por computadora en el Centro de Educación Inicial No.1 del Ministerio de Educación, se han determinado las siguientes conclusiones:

- Se determinó la situación inicial de seguridad de la institución, con la presencia de personal de seguridad quienes no poseían de medios tecnológicos que les permitan desarrollar su trabajo en forma eficiente. Se vio la necesidad de implementar un sistema de seguridad mediante cámaras de seguridad inteligentes, una para el reconocimiento de rostros y dos para la detección de personas, así como un control para de acceso para la puerta principal del establecimiento educativo.
- Con el fin de implementar sistemas de reconocimiento facial y detección de personas se seleccionó el lenguaje de programación Python de distribución libre, para la implementación del sistema. Fue elegido debido a que posee una extensa cantidad de librerías para el desarrollo de aplicaciones en Machine Learning y redes neuronales, es un lenguaje moderno que facilita la programación y provee soporte a través de comunidades en internet.
- Los elementos del sistema de video vigilancia fueron definidos y acoplados, se configuraron computadores, cámaras de seguridad digitales de alta definición, una tarjeta Arduino y una chapa magnética para el control y monitoreo de la puerta de acceso, un router para interconexión de la red, repetidores de señales y el software de detección y reconocimiento de personas. Todos ellos acoplados mediante comunicación Ethernet y USB permitieron implementar un sistema completo de seguridad de video vigilancia basado en aprendizaje automático (Machine Learning).
- Se realizó la validación del sistema y pruebas funcionales de los dispositivos instalados, en cuanto al reconocimiento de rostros se demostró su utilidad y confiabilidad en distintos niveles de iluminación, se presentaron problemas de reconocimiento con bajo nivel de luz en la noche y por exceso de luz al medio día. Respecto a accesorios en el rostro, tales como maquillaje, lentes y gorra, se obtuvo un 97% de confiabilidad en la identificación de las personas registradas en el sistema.

- Ya que se instaló un circuito de control para la chapa eléctrica de la puerta de acceso, se vio la necesidad de implementar una fuente de energía de respaldo, en caso de que no exista energía comercial para garantizar un 100% de su operatividad, además se agregó un accionamiento de emergencia para la apertura en caso de incendios o catástrofes naturales y una clave de responsabilidad que únicamente lo sabe el personal de seguridad.
- En relación con la detección de personas, se verificó que las cámaras de video están en capacidad de reconocer personas dentro del campo de visión de las cámaras. Se programó una alarma que envía una alerta ante cualquier indicio de eventualidades.



## RECOMENDACIONES

- Realizar análisis de nuevos requerimientos de seguridad en la institución educativa, en caso de la creación de nuevas zonas o de detectar puntos vulnerables, se puede implementar nuevas cámaras de video vigilancia inteligentes sin necesidad de realizar cambios significativos en el sistema.
- Realizar pruebas en nuevos sistemas embebidos que realicen el procesamiento de información e imágenes con una Raspberry pi y acoplarlos a sistemas modernos como la placa de desarrollo de proyectos de la empresa Microchip conocido como curiosity.
- Utilizar una unidad de procesamiento gráfica GPU podría permitir unificar el sistema en un solo computador de altas prestaciones, para el procesamiento de las imágenes en el ámbito de reconocimiento facial y detección de personas conforme se incremente el número de puntos de vigilancia o de usuarios autorizados.
- Realizar un mantenimiento periódico del sistema para garantizar la operatividad de las cámaras de video vigilancia. La base de datos también requiere de mantenimiento para evitar exceso de datos innecesarios en el sistema.
- Las computadoras asignadas para el sistema no deberían utilizarse con otro fin, ya que el procesamiento de imagen requiere un gran costo computacional. Por otra parte, se debería proteger a los sistemas del ataque de virus a través de dispositivos externos de seguridad o un correcto filtro web ante amenazas de ciberdelincuencia.
- Respetar las normativas y leyes vigentes en cuanto a la instalación y uso de cámaras de video vigilancia, tanto durante la instalación de la infraestructura de los sistemas de video vigilancia y del control de acceso, así como durante su puesta en marcha, los cuales están estipulados en el código orgánico de lo penal. También se debe tener en cuenta seguridades informáticas para proteger la información contenida en los dispositivos.

## BIBLIOGRAFÍA

- Aceves, F. (2013). *Sistema de video viginacia para la ciudad de México*. México: Instituto Politécnico Nacional.
- Alvarez, M. (19 de noviembre de 2003). *Qué es Python*. Obtenido de <https://desarrolloweb.com/articulos/1325.php>
- Andalucía es digital. (15 de octubre de 2018). *Deep Learning, Inteligencia artificial y Machine Learning*. Obtenido de <https://www.blog.andaluciaesdigital.es/deep-learning-inteligencia-artificial-y-machine-learning/>
- Androide Libre. (16 de julio de 2017). *Lector óptico de huellas*. Obtenido de <https://elandroidelibre.elespanol.com/2016/07/como-funciona-lector-de-huellas.html>
- Anggo, M., & Arapu, L. (2018). Face Recognition Using Fisherface Method. *Journal of Physics: Conference Series*, 1-9.
- Avitom. (16 de enero de 2017). *Cómo elegir una cámara de seguridad*. Obtenido de [www.avitom.es/elegir-una-camara-de-seguridad-segun-la-resolucion](http://www.avitom.es/elegir-una-camara-de-seguridad-segun-la-resolucion)
- Axis Communications. (2018). Estándar de compresión de video H.264. *rdsn*, 140-152.
- Bagnato, J. (4 de noviembre de 2017). *Aprende Machine Learning*. Obtenido de <https://www.aprendemachinlearning.com/principales-algoritmos-usados-en-machine-learning/>
- Berger, W. (2 de diciembre de 2018). *Deep Learning Hard Cascade Explained*. Obtenido de <http://www.willberger.org/cascade-haar-explained/>
- Caballero, F., & Reyes, M. (2017). Reconocimiento facial por el método de Eigenfaces. *Pistas educativas* 127, 66-81.
- Cabrera, L. (2017). *Diseño e implementación de una solución para el conteo de personas a partir de imágenes terrestres*. Loja: Universidad Técnica Particular de Loja.
- Calvo, D. (20 de julio de 2017). *Red neuronal convolucional CNN*. Obtenido de <http://www.diegocalvo.es/red-neuronal-convolucional/>
- Castillo, M. (16 de septiembre de 2018). *Reconocimiento de imágenes en la web utilizando tensorflow*. Obtenido de [www.medium/@ManuCastilloM/reconocimiento-de-imagenes-en-la-web](http://www.medium/@ManuCastilloM/reconocimiento-de-imagenes-en-la-web)

- Cat Colombia Solutions. (4 de marzo de 2016). *La evolución de los sistemas de seguridad electrónica*. Obtenido de <http://catcolombiasolutions.com/index.php/actualidad/76-la-evolucion-de-los-sistemas-de-seguridad-electronica>
- Catalán, D., & Paternain, D. (2017). *Detección automática de personas mediante HOG*. Navarra: Universidad Pública de Navarra.
- Centronic. (30 de julio de 2018). *Cámara turbo HD 4.0*. Obtenido de <http://blog.centronic.com.py/camara-turbo-hd-4-0-hikvision/>
- Chas, A. (16 de mayo de 2019). *Inteligencia artificial algoritmos*. Obtenido de <https://www.auraportal.com/es/tipos-de-algoritmos-de-inteligencia-artificial-y-machine-learning/>
- Datamation. (26 de agosto de 2019). *What is Artificial Intelligence*. Obtenido de <https://www.datamation.com/artificial-intelligence/what-is-artificial-intelligence.html>
- Davies, E. (2012). *Computer and Machine Vision*. USA: ELSEIER.
- Domínguez, S. (2017). *Reconocimiento facial mediante el análisis de componentes principales*. Sevilla: Universidad de Sevilla.
- EcuRed. (1 de septiembre de 2019). *Reconocimiento del iris*. Obtenido de [http://www.ecured.cu/index.php?title=Reconocimiento\\_del\\_Iris&action=history](http://www.ecured.cu/index.php?title=Reconocimiento_del_Iris&action=history)
- Elaine Rich, K. K. (1996). *Inteligencia Artificial*. España: Mac Graw - Hill.
- Espinoza, D., & Jorquera, P. (2015). *REconocimiento facial*. Valparaiso: Pontificia Universidad Católica de Valparaiso.
- Funcia, M. (2 de julio de 2018). *Historia de la cámara de seguridad*. Obtenido de <https://camarasdeseguridadparacasa.com/historia-de-la-camara-de-seguridad/>
- G., W. R. (January 1994). *Modeling and calibration of automated zoom lenses*. Carnegie Mellon University.
- González , A., & Hernández, M. (2015). Sistemas de visión computacional para la determinación de los niveles de servicios en vías peatonales de la zona céntrica de la ciudad de Montería. *Encuentro internacional de Educación en ingeniería*, 1-7.
- González, R. (2007). El test de Turing: Dos mitos, un dogma. *Revista de filosofía Volumen 63*, 37-53.
- He, K., & Zhang, X. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778.

- Hecht E., Z. A. (1989). *Óptica*. Addison-Wesley Iberoamericana.
- Intplus.es. (25 de octubre de 2019). *Video vigilancia y seguridad electrónica*. Obtenido de [http://www.superinventos.com/sistemas\\_videovigilancia.htm](http://www.superinventos.com/sistemas_videovigilancia.htm)
- Jimenez, M. (18 de enero de 2018). *Procesamiento de imágenes digitales*. Obtenido de <http://asignatura.us.es/imagendigital/Presentacion.pdf>
- Junghanss, R. (2017). Componentes y sistemas de un sistemas de CCTV. *data técnica*, 140-156.
- K.S. FU, R. G. (s.f.). *ROBOTICA Control, Detección, Visión e INteligencia*. McGraw - Hill.
- La Casa de Nicéphore Niépce. (29 de julio de 2019). *Daguerre y la invención de la fotografía*. Obtenido de <http://photo-museum.org/es/daguerre-invencion-foto/>
- Lacort, J. (21 de agosto de 2017). *Las claves de los sistemas de reconocimiento facial*. Obtenido de <https://www.xataka.com/seguridad/las-claves-de-los-sistemas-de-reconocimiento-facial-cual-es-su-verdadero-nivel-de-seguridad>
- LeCun, Y., & Boser, J. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Comput*, 541-551.
- Linde, H., & Schweizer, I. (2019). A White Paper on the Future of Artificial Intelligence. *ResearchGate*, 1-10.
- López, M. (2017). *Identificación de sistemas no lineales con redes neuronales convolucionales*. México: Centro de investigación y de estudios avanzados del Instituto Politécnico Nacional.
- López, R. (2 de agosto de 2016). *REdes neuronales convolucionales con TensorFlow*. Obtenido de <https://relopezbriega.github.io/blog/2016/08/02/redes-neuronales-convolucionales-con-tensorflow/>
- MathWorks. (10 de diciembre de 2019). *Image processing and computer vision*. Obtenido de <https://la.mathworks.com/solutions/image-video-processing/object-recognition.html>
- MathWorks. (12 de octubre de 2019). *Redes neuronales convolucionales*. Obtenido de <https://la.mathworks.com/solutions/deep-learning/convolutional-neural-network.html>
- Merchán, F., & Galeano, S. (2014). Mejoras en el entrenamiento de esquemas de detección de sonrisas basados en Adaboost. *Revistas Académicas UTP*, 17-30.
- Mora, R. (10 de diciembre de 2018). *Training image classifiers using tensorflow object detection API*. Obtenido de [www.geeksforgeeks.org/ml-training-image-classifier-using-tensorflow-object-detection-api/](http://www.geeksforgeeks.org/ml-training-image-classifier-using-tensorflow-object-detection-api/)

- Morales, J. T. (2008). *Inteligencia Artificial Técnicas métodos y Aplicaciones*. España: Mc Graw Hill.
- Morling Global. (22 de septiembre de 2018). *k-Nearest Neighbour kNN*. Obtenido de <https://bigishere.wordpress.com/2018/09/22/k-nearest-neighbour-knn/>
- Mosquera, V., & Romero, E. (2016). *Diseño de un software piloto de reconocimiento facial*. Bárbula: Universidad de Carabobo.
- Mut, V., & Romero, L. (2016). *Procesamiento digital de imágenes*. San Juan, Puerto Rico: Universidad Nacional de San Juan.
- Nilson, N. J. (1986). *Principle of Artificial Inteligence*. Morgan Kaufman.
- Pardo, C. (23 de abril de 2018). *Histograma de gradientes orientados HOG*. Obtenido de Visión por computador: <https://carlosjuliopardoblog.wordpress.com/2017/12/04/hog-histograma-de-gradientes-orientados/>
- Python Awesome. (25 de septiembre de 2018). *VGGFace2 Dataset for Face Recognition*. Obtenido de <https://pythonawesome.com/vggface2-dataset-for-face-recognition/>
- Querejeta, P. (2015). *Procesamiento digital de imágenes*. Bahía Blanca, Argentina: Universidad Nacional del Sur.
- Recuero, P. (16 de noviembre de 2017). *Inteligencia artificial*. Obtenido de <https://empresas.blogthinkbig.com/que-algoritmo-elegir-en-ml-aprendizaje/>
- Rosembrock, A. (3 de abril de 2017). *Facial landmarks with dlib, OpenCV and python* . Obtenido de <https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/>
- Ruiz, J., & Monroy, J. (2018). Tutorial para el reconocimiento de objetos basado en características empleando herramientas Python. *Actas de las XXXIX Jornadas de Automática, Badajoz*, 998-1005.
- Saiz, A. (16 de Septiembre de 2008). *Perspectiva lineal en Brunelleschi*. Obtenido de [www.uv.es/mahiques/ENCICLOPEDIA/BRUNELLESCHI/perspectiva.pdf](http://www.uv.es/mahiques/ENCICLOPEDIA/BRUNELLESCHI/perspectiva.pdf)
- Sancho, F. (14 de diciembre de 2009). *Redes neuronales: Una visión superficial*. Obtenido de <http://www.cs.us.es/~fsancho/?e=72>
- Security Set. (5 de diciembre de 2016). *Compresión de video*. Obtenido de <http://www.blog.securityset.cl/compresion-video/>

Seguridad TV. (12 de abril de 2018). *Cámaras de seguridad, Sistemas de videovigilancia.*

Obtenido de <https://www.seguridadtv.com/>

Smart Panel. (12 de julio de 2019). *Qué es Deep Learning?* Obtenido de

<https://www.smartpanel.com/que-es-deep-learning/>

Tecnoseguro. (23 de junio de 2018). *Qué es un sistema de control de acceso?* Obtenido de

<https://www.tecnoseguro.com/faqs/control-de-acceso/que-es-un-control-de-acceso>

Vásquez, M. (2014). *Sistema de reconocimiento facial mediante técnicas de visión*

*tridimensional*. Guanajuato: Centro de investigaciones en óptica, A.C. .

Vernon, D. (1991). *Machine vision: automated visual inspection and robot vision*. Englewood

Cliffs: Prentice Hall.

Winston, P. H. (1992). *Artificial Intelligence*. Addison - Wesley.

## ANEXO 1

### ESTUDIO DE INGENIERÍA PARA DETERMINAR EL COSTE DE EJECUCIÓN DE UN SISTEMA CCTV MEDIANTE VISIÓN POR COMPUTADOR PARA EL CENTRO DE EDUCACIÓN INICIAL NUMERO 1 DEL MINISTERIO DE EDUCACIÓN

#### OBJETIVO:

Elaborar el presupuesto para la instalación de un CCTV mediante visión por computadora para el CEI N 1

El análisis de costos promueve el conocimiento de la estructura de costos involucrada con los productos y servicios del proyecto de VIDEO VIGILANCIA MEDIANTE VISIÓN POR COMPUTADOR.

Se analizan los elementos específicos de los módulos que conforman el sistema así como el trabajo requerido y los gastos generales.

A continuación, se detalla los costos de los materiales y demás insumos empleados en el proyecto.

#### CUADRO DE PRECIOS DE MATERIALES:

MODELO	UD	DESCRIPCIÓN	CANTIDAD	PRECIO	TOTAL
<b>MATERIALES PRINCIPALES</b>					
	Unidades	LAPTOP TOSHIBA	1	1200	1200
	Unidades	GPU LENOVO i7 8va generación con procesador RADEON	1	1400	1400
WIC20	Unidades	CAMARA IP EPCOM CUBICA	1	131.06	131.06
AE-CP315	Unidades	COOLING PAD	1	12.050	12.050
C270HD	Unidades	WEB CAM LOGITECH	1	80.51	80.51
56262	Unidades	CERRADURA RIM LOCK SILK	1	37.5	37.5
	Unidades	CONTROLADOR DE CARGA SOLAR	1	15.89	15.89
NAXOS 500	Unidades	SWITCH FAST ETHERNET	1	11.57	11.57
AE-HUB	Unidades	HUB USB 2.0 DE 4 PUERTOS	1	4.86	4.86
	Unidades	CARGADOR ADAPTADOR 12 V 2A	3	4.28	4.28
		ARDUINO UNO	1	5.0	5.0
	Unidades	MODULO RELE DE UN CANAL	1	2.32	2.32
		MÓDULO MP3	1	5.0	5.0
50Nk	Unidades	SENSOR INFRAROJO	1	15.18	15.18
	Unidades	SENSOR MAGNÉTICO	1	2.15	2.15
DIR 809	Unidades	ROTEADOR WI-FI ROUTER DLINK AC 750 DUAL BAND	1	36.5	36.5

	Unidades	LCD DISPLAY 5 PULGADAS	1	66.96	66.96
	Unidades	BATERIA SECA RECARGABLE DE 12 V	1	23.21	23.21
T12HD	Paquetes	AMARRAS 30 CM	1	4.43	4.43
<b>MANO DE OBRA</b>					
	Horas	OFICIAL	30	4	120
	Horas	AYUDANTE	30	2	60
<b>MEDIOS AUXILIARES</b>					
		SEÑAL ÉTICA DE SEGURIDAD INDUSTRIAL	1	2.5	2.5
	Metros	EXTENSIÓN DE CABLE USB	1	26.10	26.10
	Metros	EXTENSIÓN DE CABLE HDMI	3	1.41	4.24
	Metros	EXTENSIÓN USB PARA ARDUINO	1	5.36	5.36
	Caja	CABLE UTP CATEGORÍA 5E 100M	1	43.33	43.33
	Rollo	CABLE AWG 14	1	45.54	45.54
	Metros	CANALETAS 24/14MM ADHESIVA	20	1.52	27.93
	Metros	CANALETAS 48/28 MM ADHESIVA	2	5.6	11.20
	Metros	CABLE INSTALACIÓN GPT 18 AWG	10	0.21	2.10
	Unidades	CAJETÍN	2	17.0	34
	Unidades	TOMAS SOBREPUESTOS	2	3.31	6.62
	Metros cuadrados	CORTE Y GRABADO ACRILICO DE 3 mm	(1/8) m <sup>2</sup>	22	22
	Unidades	CONECTOR DB15	1	3.5	3.5
	Unidades	TACOS F8	16	0.04	0.64
	Unidades	TACO FISHER F6	20	0.02	0.40
	Unidades	TORNILLOS 6"3/4	20	0.02	0.40
	Unidades	TORNILLOS VARIOS	25	0.02	0.50
	Unidades	BORNERA PIN INDUSTRIAL	1	1.7	1.7
	Paquetes	CUCHILLAS	1	1.29	1.29
	Unidades	CRIMPADORA DE IMPACTO	1	3.57	3.57
	Unidades	JACK RJ45 CAT 6	1	3.48	3.48
	Unidades	UNION HDMI HEMBRA	1	3.20	3.20
<b>SUBTOTAL</b>					<b>3488.07</b>
<b>IVA</b>					<b>418.56</b>
<b>TOTAL</b>					<b>3906.63</b>



## ANEXO 2

### ALGORITMO KNN DE ENTRENAMIENTO DEL CONJUNTO DE DATOS

**Nombre:** Train.py

```
import math
from sklearn import neighbors
import os
import os.path
import pickle
import face_recognition
from face_recognition.face_recognition_cli import image_files_in_folder

train_dir = "Usuarios"
verbose = True
n_neighbors = None
knn_algo = 'ball_tree' #algoritmo para establecer vecinos mas cercanos
model_save_path = 'faces_encodings.pickle'

X = []
y = []

print('[INFO] Iniciando Entrenamiento...')
for class_dir in os.listdir(train_dir):
    if not os.path.isdir(os.path.join(train_dir, class_dir)):
        continue
    for img_path in image_files_in_folder(os.path.join(train_dir, class_dir)):
        print(img_path)
        image = face_recognition.load_image_file(img_path)
        face_bounding_boxes = face_recognition.face_locations(image)
        if len(face_bounding_boxes) != 1:
            if verbose:
                print("Image {} not suitable for training: {}".format(img_path, "Didn't find a face" if
len(face_bounding_boxes) < 1 else "Found more than one face"))
            else:
                X.append(face_recognition.face_encodings(image, known_face_locations=face_bounding_boxes)[0])
                y.append(class_dir)
if n_neighbors is None:
    n_neighbors = int(round(math.sqrt(len(X))))
    if verbose:
        print("[INFO] Seleccionado n_neighbors automáticamente", n_neighbors)
knn_clf = neighbors.KNeighborsClassifier(n_neighbors=n_neighbors, algorithm=knn_algo, weights='distance')
knn_clf.fit(X, y)
print("K utilizado: ",nneighbors)
from sklearn.metrics import accuracy_score
acc_knn = accuracy_score(y, knn_clf.predict(X))
print('KNN training accuracy = ' + str(100*acc_knn) + '%')
if model_save_path is not None:
    with open(model_save_path, ) as f:
        pickle.dump(knn_clf, f)
    print('[INFO] Entrenamiento Realizado!')
```

## ANEXO 3

### ALGORITMO DE RECONOCIMIENTO DE ROSTROS

**Nombre:** gui\_knn\_final.py

```
import tkinter as tk
import cv2
import numpy as np
from PIL import Image, ImageTk
import dlib
import pickle
import os
import face_recognition
import math
from sklearn import neighbors
import os.path
import subprocess
import time
import datetime
Rectangle = np.zeros((480,640,3), np.uint8)
Rectangle[:,:] = [0,0,0]
ROI = np.zeros((360,320,3), np.uint8)
ROI[:,:] = [0,0,0]

contador = 0
x2 = 0
x1 = 0

actualizador = False
contador_actualizador = 0
contador_fotos = 0
detector = dlib.get_frontal_face_detector()
knn_clf = pickle.loads(open("faces_encodings.pickle", "rb").read())
distance_threshold = 0.6# .06
class WebcamVideoStream :
    def __init__(self, src = 'rtsp://admin:@xxx.xxx.xxx.xxx', width = 320, height = 240) :
        self.stream = cv2.VideoCapture(src)
        (self.grabbed, self.frame) = self.stream.read()

        self.started = False

    def start(self) :
        if self.started :
            print ("already started!!")
            return None
        self.started = True
        self.thread.start()
        return self

    def update(self) :
        global streaming
        while self.started :
            (grabbed, frame) = self.stream.read()
            self.read_lock.acquire()
            self.read_lock.release()

    def read(self) :
        self.read_lock.acquire()
```

```

    frame = self.frame.copy()
    #frame = cv2.resize(frame,None, fx=0.5,fy=0.5,interpolation = cv2.INTER_CUBIC)
    return frame

def stop(self) :
    self.started = False
    self.thread.join()

def __exit__(self, exc_type, exc_value, traceback) :
    self.stream.release()

class WebcamVideoStream :
def __init__(self, src = rtsp://admin:@xxx.xxx.xxx.xxx, width = 640, height = 480) :
    self.stream = cv2.VideoCapture(src)
    (self.grabbed, self.frame) = self.stream.read()

def start(self) :
    self.thread = Thread(target=self.get, args=())
    return self
def get(self) :
    global streaming
    while not self.stopped
        if not self.grabbed:
            streaming = False
            print("NO HAY CAMARA")
        else:
            (self.grabbed, self.frame) = self.stream.read()
            print("aqui estoy")

def stop(self) :
    global streaming
    self.stopped = True
    streaming = False
def __exit__(self, exc_type, exc_value, traceback) :
    self.stream.release()
def Main_Folder(path):
    try:
        if not os.path.exists(path):
            print ("Exitosa creacion de directorio %s " % path)
        else:
            print ("El directorio %s ya existe" % path)

    except OSError:
        print ("Creacion de directorio %s fallo" % path)
def createFolder(directory):
    if not os.path.exists(directory):
        os.makedirs(directory)

    except OSError:
        print ('Creacion de directorio %s fallo' % directory)

def recuadro(color):
    cv2.line(suma,(160,40),(220,40),color,7,cv2.LINE_AA)
    cv2.line(suma,(160,400),(160,340),color,7,cv2.LINE_AA)

    cv2.line(suma,(480,40),(420,40),color,7,cv2.LINE_AA)
    cv2.line(suma,(480,400),(420,400),color,7,cv2.LINE_AA)

    subprocess.Popen(['C:\Program Files (x86)\Ezviz Studio\EzvizProtect.exe'],shell=True)

def show_CCTV():

```

```

global streaming
if streaming == True:
    vs.stop()
def start_stream():
    global streaming
    num = Entry1.get()
    if streaming == False:
        vs = WebcamVideoStream().start()
        show_frame()
    else:
        print("estoy corriendo")

def show_frame():
    global ROI
    global Rectangle
    global detector
    global knn_clf
    global suma
    global contador
    global contador_actualizador
    global x2
    global x1
    global deteccion
    global distance_threshold
    global Nombre_subFolder
    global contador_fotos

    img = vs.frame
    if vs.stopped == True:
        print("xomo ESTA ALIVE",vs.thread.is_alive())

    else:
        print("ENTRE")
        #btn.configure(state=DISABLED)
        """
        Para mostra imagen a color en el label es necesario que el orden de los
        plano sea RGB: cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        """
        ROI = img[40:400,160:480]
        copia_ROI = ROI.copy()
        small_ROI = cv2.resize(ROI, (0, 0), fx=0.25, fy=0.25)
        image_RGB = small_ROI[:, :, ::-1]
        face_landmarks_list = face_recognition.face_landmarks(image_RGB,model="small")
        print("dfadfasfads",len(face_landmarks_list))
        for face_landmarks in face_landmarks_list:

            for facial_feature in face_landmarks.keys():

                pts = face_landmarks[facial_feature]
                if facial_feature == "right_eye":
                    (x1,y1) = pts[0]

                if facial_feature == "left_eye":
                    (x2,y2) = pts[0]

            for (x, y) in pts:
                cv2.circle(copia_ROI, (4*x, 4*y), 3, (0, 0, 255), -1)

        distancia =abs(x2-x1)
        print("distancia coordenada x:", distancia )

```

```

if rostros == 0:
    distancia = 0
if distancia > 16: # distancia entre puntos de ojos
    cv2.circle(copia_ROI, (4*x1, 4*y1), 6, (0, 255, 255), -1)
    contador = contador + 1

else:
    contador=0

print("contador: ",contador)

suma = cv2.addWeighted(Rectangle,0.5,img,0.5,0)
recuadro(color)
suma[40:400,160:480] = copia_ROI #suma 640x480
if actualizador == True:
    contador_actualizador = contador_actualizador + 1
    if contador_actualizador == 40:
        Frame2.configure(background="#d9d9d9") # color gris
        imagen_label_2 = ImageTk.PhotoImage(Image.open('user.png'))
        lbl4.configure(text="")
        lbl5.configure(text="")

        lbl2.imgtk = imagen_label_2# esto fue definitivo
        actualizador = False
        contador_actualizador = 0
if contador == 20 and deteccion == False: # esperando que el usuario se ubique bien en recuadro
    cv2.imwrite("roi.jpg",copia_ROI)
    deteccion = True

if deteccion == True:
    # -a) Localizacion de rostro en image_RGB (ROI al 25% con canales invertidos)
    face_locations = face_recognition.face_locations(image_RGB)
    if len(face_locations) == 1: # se encontro 1 solo rostro
        face_encodings = face_recognition.face_encodings(image_RGB, face_locations)

        # Use the KNN model to find the best matches for the test face
        closest_distances = knn_clf.kneighbors(face_encodings, n_neighbors=1)
        are_matches = [closest_distances[0][i][0] <= distance_threshold for i in range(len(face_locations))]
        predictions = [(pred, loc) if rec else ("unknown", loc) for pred, loc, rec in
zip(knn_clf.predict(face_encodings),face_locations, are_matches)]
        print(predictions)

        if are_matches[0] == True: # si se presento coincidencia
            # h) Mostramos la imagen correspondiente a nombre_00.jpg para el usuario reconocido

            result = [x.strip() for x in name.split('-')]
            name_user = result[0]
            rol_user = result[1]

            user = cv2.imread("Usuarios/" + name + "/1.jpg")
            user_resized = cv2.resize(user,(200,200))
            img_2 = cv2.cvtColor(user_resized, cv2.COLOR_BGR2RGB)
            lbl2.configure(image=imgtk_2)
            if contador_fotos >= 1 and contador_fotos <= 3 and actualizador == False:
                path_save = path + Nombre_subFolder
                ahora = datetime.datetime.now()
                hora = str(ahora.hour)
                minuto = str(ahora.minute)
                segundo = str(ahora.second)
                nombre_imagen = name_user+'-' + hora+'h-'+minuto+'m-'+segundo+'s+'.jpg'
                cv2.imwrite(os.path.join(path_save , nombre_imagen), copia_ROI_2)

```

```

        if contador_fotos == 3:

            contador_fotos = 0
            actualizador = True # para mostrar nuevamente portada despues de un tiempo
        else: # usuario desconocido
            Frame2.configure(background="#d83529") # color rojo
            user_resized = cv2.resize(copia_ROI,(200,225))
            img_2 = cv2.cvtColor(user_resized, cv2.COLOR_BGR2RGB)
            lbl2.imgtk = imgtk_2# esto fue definitivo
            lbl2.configure(image=imgtk_2)
            deteccion = False
            actualizador = True # para mostrar nuevamente portada despues de un tiempo

    deteccion = False
    contador=0
    distancia = 0 # adiconado para solucionar dibujo de circulos amarillos continuos
    img_1 = cv2.cvtColor(imagen_label, cv2.COLOR_BGR2RGB)
    im = Image.fromarray(img_1)
    imgtk = ImageTk.PhotoImage(image=im)
    lbl.configure(image=imgtk)
    window.after(10, show_frame)
window=tk.Tk()
btn=tk.Button(window, text="Start")
#btn.place(relx=0.026, rely=0.894, height=34, width=157)
btn.place(relx=0.013, rely=0.894, height=54, width=54)
btn.configure(activebackground="#ecec")
btn.configure(foreground="#000000")
btn.configure(highlightbackground="#d9d9d9")
btn.configure(highlightcolor="black")
btn.configure(pady="0")
img_boton_2 = ImageTk.PhotoImage(Image.open('2.png'))
btn.config(image=img_boton_2 )
btn.configure(command=start_stream)

btn1=tk.Button(window, text="Open CCTV")
btn1.place(relx=0.767, rely=0.894, height=34, width=157)
btn1.configure(activebackground="#ecec")
btn1.configure(highlightbackground="#d9d9d9")
btn1.configure(highlightcolor="black")
btn1.configure(pady="0")
btn1.configure(command=show_camaras)

Frame1 = tk.Frame(window)
Frame1.place(relx=0.013, rely=0.067, relheight=0.809, relwidth=0.563)
Frame1.configure(relief='groove')
Frame1.configure(highlightcolor="black")
Frame1.configure(width=445)

lbl=tk.Label(Frame1)
lbl.place(relx=0.045, rely=0.042, height=440, width=400)
lbl.configure(activebackground="#f9f9f9")
lbl.configure(activeforeground="black")
lbl.configure(highlightbackground="#d9d9d9")
lbl.configure(highlightcolor="black")
lbl.configure(text=""Label"")

Frame2 =tk.Frame(window)
Frame2.place(relx=0.588, rely=0.067, relheight=0.143, relwidth=0.39)
Frame2.configure(relief='groove')
Frame2.configure(borderwidth="2")

```

```
Frame2.configure(relief="groove")
Frame2.configure(highlightbackground="#d9d9d9")

Frame3 = tk.Frame(window)
Frame3.place(relx=0.588, rely=0.219, relheight=0.481, relwidth=0.39)
Frame3.configure(highlightbackground="#d9d9d9")
Frame3.configure(highlightcolor="black")
Frame3.configure(width=305)

lbl2=tk.Label(Frame3)
lbl2.place(relx=0.164, rely=0.14, height=200, width=200)
lbl2.configure(activebackground="#f9f9f9")
lbl2.configure(activeforeground="black")
lbl2.configure(background="#d9d9d9")
lbl2.configure(highlightbackground="#d9d9d9")
lbl2.configure(highlightcolor="black")
lbl2.configure(text="Label")
img = ImageTk.PhotoImage(Image.open('user.png'))
lbl2.configure(image = img)

lbl3=tk.Label(window)
lbl3.place(relx=0.588, rely=0.698, height=27, width=114)
lbl3.configure(background="#3529d8")
lbl3.configure(disabledforeground="#a3a3a3")
lbl3.configure(foreground="#ffffff")
lbl3.configure(highlightbackground="#d9d9d9")
lbl3.configure(highlightcolor="black")
lbl3.configure(text="USERNAME")
lbl3.configure(width=114)

Frame4 = tk.Frame(window)
Frame4.place(relx=0.588, rely=0.742, relheight=0.135, relwidth=0.39)
Frame4.configure(relief='groove')
Frame4.configure(borderwidth="2")
Frame4.configure(highlightbackground="#d9d9d9")
Frame4.configure(highlightcolor="black")
Frame4.configure(width=305)

lbl4.place(relx=0.098, rely=0.125, height=21, width=254)
lbl4.configure(activebackground="#f9f9f9")
lbl4.configure(activeforeground="black")
lbl4.configure(background="#d9d9d9")
lbl4.configure(highlightbackground="#d9d9d9")
lbl4.configure(highlightcolor="black")
lbl4.configure(font="-family {Segoe UI} -size 15 -weight bold")

lbl5=tk.Label(Frame4)
lbl5.place(relx=0.098, rely=0.525, height=21, width=254)
lbl5.configure(activebackground="#f9f9f9")
lbl5.configure(activeforeground="black")
lbl5.configure(highlightbackground="#d9d9d9")
lbl5.configure(highlightcolor="black")
lbl5.configure(font="-family {Segoe UI} -size 9 -weight bold")
lbl5.configure(justify='left')

Entry1 = tk.Entry(window)
Entry1.place(relx=0.32, rely=0.894, height=34, relwidth=0.248)
Entry1.configure(font="-family {Courier New} -size 14")
Entry1.configure(foreground="#000000")
Entry1.configure(insertbackground="black")
Entry1.configure(justify='center')
```

```
Entry1.configure(width=194)

Button2 = tk.Button(window)
Button2.place(relx=0.614, rely=0.894, height=54, width=54)
Button2.configure(activebackground="#ecec")
Button2.configure(activeforeground="#000000")
Button2.configure(background="#d9d9d9")
Button2.configure(highlightbackground="#d9d9d9")
Button2.configure(highlightcolor="black")
Button2.configure(pady="0")
Button2.configure(text="Button")
Button2.configure(width=57)
img_boton_2 = ImageTk.PhotoImage(Image.open("2.png"))
#img_boton_2 = PhotoImage(file="2.png") # make sure to add "/" not "\"
Button2.config(image=img_boton_2 )

window.title('FACE RECOGNITION SYSTEM')# titulo de ventana
window.resizable(False,False) # crea ventana estatica

path = 'C:/REGISTROS_ACCESO/'
Main_Folder(path)

now = datetime.datetime.now()
createFolder(path + Nombre_subFolder +'/')

window.mainloop()
if streaming == True:
    vs.stop()
    #vs.stream.release()
```



## ANEXO 4

### SOFTWARE DE CONTROL DE LA TARJETA ARDUINO

```
//Librerías:
//Librerías:
#include <SoftwareSerial.h>
#include <Password.h>
#include <Keypad.h>
#include <DFRobotDFPlayerMini.h>
//Sensors Const:
const int FirePin = 3; //Sistema de Incendios
const int OpenDoor = 9; //Señal Abrir Puerta
const int DoorPin = 14; //Sensor Puerta Abierta
const int PresencePin = 15; //Sensor de Presencia
bool aux = false;
bool aux3 = true;
bool aux4 = true;

SoftwareSerial audioSerial(10, 11); // RX, TX

Password password = Password( "159" );

const byte rows = 4; //4 Filas
const byte cols = 3; //3 Columnas
char keys[rows][cols] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
};
byte rowPins[rows] = {16, 17, 18, 19};
byte colPins[cols] = {8, 7, 6};
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, rows, cols );

void setup() {
  pinMode(PresencePin, INPUT_PULLUP);
  pinMode(FirePin, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(FirePin), ManualOpen, RISING);
  pinMode(OpenDoor, OUTPUT);
  pinMode(LED_BUILTIN, OUTPUT);
  //Señales Voltaje
  pinMode(5, OUTPUT);
  digitalWrite(5, LOW);
  digitalWrite(OpenDoor, HIGH);
  Serial.begin(9600);
  audioSerial.begin(9600);
  Serial.println("INICIO:");
  if (!mp3audio.begin(audioSerial)) {
    Serial.println("Error de reproductor:");
    Serial.println("1.Revise la conexión!");
    Serial.println("2.Inserte la tarjeta SD!");
    while(true);
  }
  mp3audio.volume(20);
  keypad.addEventListener(keypadEvent);
}

void loop() {
  keypad.getKey
  if (digitalRead(DoorPin) == LOW)
```

```

if (aux2 == false) {
  mp3audio.pause();
  digitalWrite(LED_BUILTIN, LOW);
}

if (digitalRead(PresencePin) == LOW) { //Da mensaje al detectar persona
  if (aux3 == true){
    mp3audio.playMp3Folder(1);
    aux3 = false;
  }
}
if (aux3 == false) {
  if (digitalRead(PresencePin) == HIGH) {
    aux3 = true;
  }
}
if ((Serial.available()>0)) {
  aux3 = true;
  aux4 = true;
  char option = Serial.read
  if (option == '1') {
    mp3audio.playMp3Folder(2)
    delay(3250);
  }
  else{
    if (aux4 == true) {
      aux4 = false;
      mp3audio.playMp3Folder(4);
    }
  }
}
}
else {
  puertaopen();
}
}

void puertaopen (){
  if (Serial.available()>0){
    char auxs = Serial.read();
  }
  if (aux2 == true){
    digitalWrite(LED_BUILTIN, HIGH);
    mp3audio.playMp3Folder(5);
    aux2 = false;
  }
  aux3 = true;
  aux4 = true;
}

void ManualOpen (){
  if (aux == false){
    aux = true;
  }
}
void AutoOpen (){
  digitalWrite(OpenDoor, LOW.
  delay(500);
  digitalWrite(OpenDoor, HIGH);
}

void keypadEvent(KeypadEvent eKey) {

```

```
if(keypad.getState() == PRESSED){
  Serial.println(eKey);
  switch (eKey){
    case '*': checkPassword(); break;
    default: password.append(eKey);
  }
}
void checkPassword(){
  if (password.evaluate()){
    AutoOpen();
  }
  password.reset();
}
```

## ANEXO 5

### ALGORITMO DE ETIQUETAS DE RECONOCIMIENTO DE OBJETOS

**Archivo:** cocolabels.py

```
LABEL_MAP = {
    0: "unlabeled",
    1: "persona",
    2: "bicycle",
    3: "car",
    4: "motorcycle",
    5: "airplane",
    6: "bus",
    7: "train",
    8: "truck",
    9: "boat",
    10: "traffic"
}
```

El algoritmo de detección de objetos se ejecuta permanentemente en el computador, realiza el reconocimiento de personas a través de dos cámaras IP.

- **Algoritmo de detección de objetos**

**Nombre:** SSDInceptionTensorflow\_THREAD.py

```
"""
Python 3.6 + OpenCV 3.4.2
"""
from threading import Thread, Lock
import cv2
import sys
import numpy as np
import time
import imutils

FROZEN_GRAPH = "ssd_inception_v2_coco.pb"
PB_TXT = "ssd_inception_v2_coco.pbtxt"
SIZE = 300

from coco_labels import LABEL_MAP

def run(img):
    cvNet = cv2.dnn.readNetFromTensorflow(FROZEN_GRAPH, PB_TXT)

    rows = img.shape[0]
    cvNet.setInput(cv2.dnn.blobFromImage(img, 1.0/127.5, (SIZE, SIZE), (127.5, 127.5, 127.5), swapRB=True,
crop=False))
    cvOut = cvNet.forward()

    for detection in cvOut[0,0,:,:]:
        score = float(detection[2])
        if score > 0.3:
            left = detection[3] * cols
            right = detection[5] * cols
            bottom = detection[6] * rows

            if LABEL_MAP[int(detection[1])] == "persona" :#or LABEL_MAP[int(detection[1])] == "truck":

                cv2.rectangle(img, (int(left), int(top)), (int(right), int(bottom)), (23, 230, 210), thickness=2)
```

```

        cv2.putText(img, LABEL_MAP[int(detection[1])], (int(left), int(top)), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(0, 255, 0))

        #print("detection",LABEL_MAP[int(detection[1])])

    return img

class WebcamVideoStream :
    def __init__(self, src = 0, width = 640, height = 480) : # modificar la resolucio de imagen como quiero
    #def __init__(self, src = 'rtsp://admin:admin123@192.168.1.108', width = 320, height = 240) :
        self.stream = cv2.VideoCapture(src)
        (self.grabbed, self.frame) = self.stream.read()

        self.started = False
        self.read_lock = Lock()

    def start(self) :
        if self.started :
            print ("already started!!")
            return None
        self.started = True
        self.thread.start()
        return self

    def update(self) :
        while self.started :
            (grabbed, frame) = self.stream.read()
            self.grabbed, self.frame = grabbed, frame
            self.read_lock.release()

    def read(self) :
        self.read_lock.acquire()
        #frame = cv2.resize(frame,None,fx=0.5,fy=0.5,interpolation = cv2.INTER_CUBIC)
        self.read_lock.release()
        return frame

    def stop(self) :
        self.started = False
        if self.thread.is_alive():
            self.thread.join()

    def __exit__(self, exc_type, exc_value, traceback) :
        self.stream.release()

if __name__ == "__main__" :
    vs = WebcamVideoStream().start()
    while True :
        frame = vs.read()
        image = run(frame)
        cv2.imshow("detections", image)

        tecla = cv2.waitKey(1) & 0xFF
        if tecla == 27:
            break

    vs.stop()
    cv2.destroyAllWindows()

```

## ANEXO 5

### ALGORITMO SERVIDOR CCTV

**Archivo:** cocolabels.py

```
import numpy as np
import cv2

Final = cv2.imread('1.png')
image_hub = imagezmq.ImageHub()

while True:
    image_name, image = image_hub.recv_image()

    if image_name == 'PC_CCTV_DETECCION_PERSONAS':

        #cv2.imshow(image_name, image)
        cv2.imshow('SERVIDOR', Final)
        image_hub.send_reply(b'OK')
        if tecla == 27:
            break
```

## ANEXO 5

### ALGORITMO SERVIDOR CCTV

**Archivo:** cocolabels.py

```
import numpy as np
import cv2
import sys
import imutils

from coco_labels import LABEL_MAP

def run(img):
    cvNet = cv2.dnn.readNetFromTensorflow(FROZEN_GRAPH, PB_TXT)

    rows = img.shape[0]
    cvNet.setInput(cv2.dnn.blobFromImage(img, 1.0/127.5, (SIZE, SIZE), (127.5, 127.), swapRB=True, crop=False))
    cvOut = cvNet.forward()

    for detection in cvOut[0,0,:,:]:
        score = float(detection[2])
        if score > 0.3:
            left = detection[3] * cols
            right = detection[5] * cols
            bottom = detection[6] * rows

            if LABEL_MAP[int(detection[1])] == "persona" :#or LABEL_MAP[int(detection[1])] == "truck":

                cv2.putText(img, LABEL_MAP[int(detection[1])], (int(left), int(top)), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(0, 255, 0))

    return img

class WebcamVideoStream :
    self.stream = cv2.VideoCapture(src)
    (self.grabbed, self.frame) = self.stream.read()
    self.read_lock = Lock()

    def start(self) :
        if self.started :
            print ("already started!!")
            return None
        self.started = True
        self.thread.start()
        return self

    def update(self) :
        while self.started :
            (grabbed, frame) = self.stream.read()
            self.read_lock.acquire()
            self.read_lock.release()

    def read(self) :
        self.read_lock.acquire()
        self.read_lock.release()
        return frame
```

```
def stop(self) :
    if self.thread.is_alive():
        self.thread.join()

def __exit__(self, exc_type, exc_value, traceback) :
    self.stream.release()

if name=="main" :

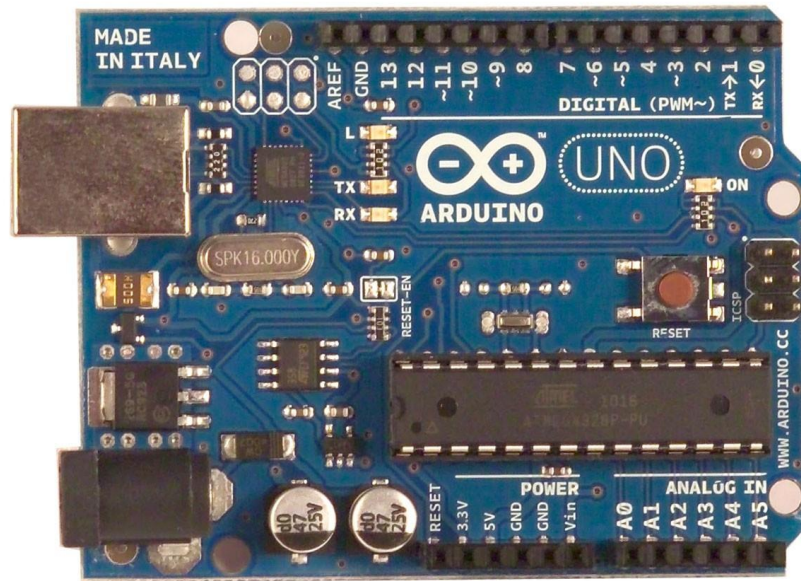
    vs = WebcamVideoStream().start()
    while True :
        frame = vs.read()
        cv2.imshow("detection", image)

        tecla = cv2.waitKey(1) & 0xFF
        if tecla == 27:
            break

    vs.stop(0)
    cv2.destroyAllWindows()
```



# Arduino UNO



## Product Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version

1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

## Index

Technical Specifications

Page 2

How to use Arduino  
Programming Environment, Basic Tutorials

Page 6

Terms & Conditions

Page 7

Environmental Policies  
half sqm of green via Impatto Zero®

Page 7

# Technical Specification

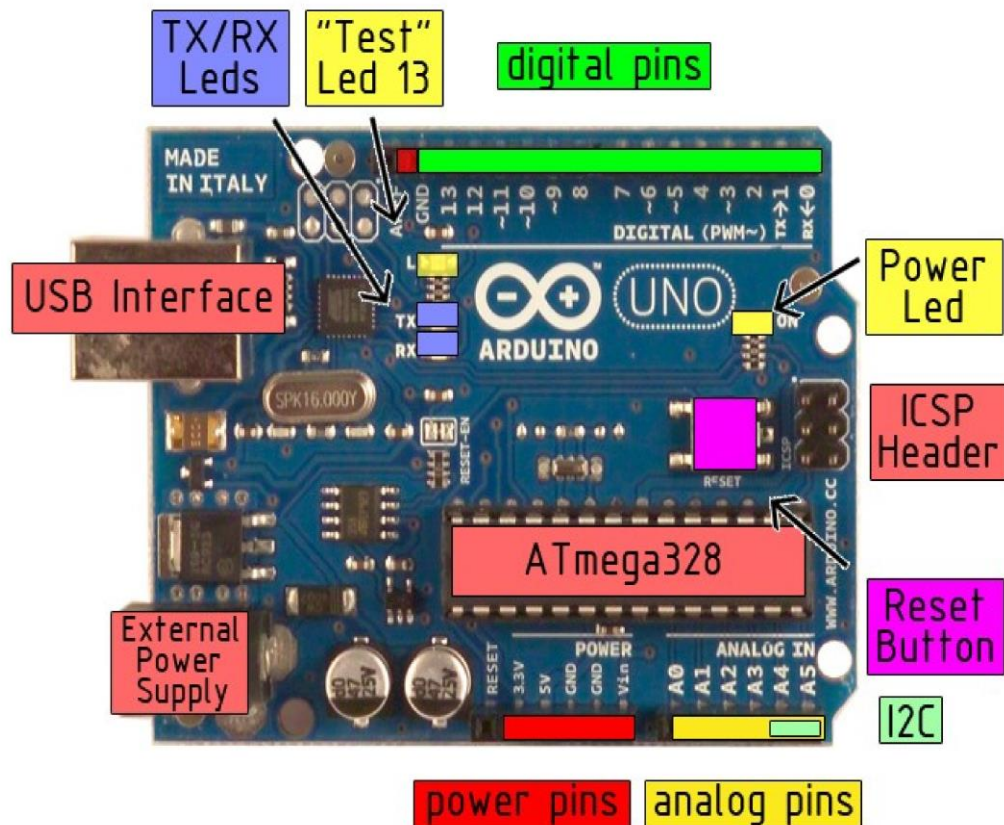


EAGLE files: [arduino-duemilanove-uno-design.zip](#) Schematic: [arduino-uno-schematic.pdf](#)

## Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended) 7-12V Input	
Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

## the board



## Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

## Memory

The Atmega328 has 32 KB of flash memory for storing code (of which 0,5 KB is used for the bootloader); It has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

## Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **I<sup>2</sup>C: 4 (SDA) and 5 (SCL).** Support I<sup>2</sup>C (TWI) communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and Atmega328 ports](#).

## Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an \*.inf file is required..

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to- serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. To use the SPI communication, please see the ATmega328 datasheet.

## Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega8U2 firmware source code is available . The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).

## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

## USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

## Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



**radiospares**

**RADIONICS**



# How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platoform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](#) for the latest instructions.  
<http://arduino.cc/en/Guide/HomePage>

## Linux Install

## Windows Install

## Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

## Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>  
Arduino-  
0017>Examples>  
Digital>Blink**

Once you have your skecth you'll see something very close to the screenshot on the right.

In **Tools>Board**

select Now you

have to go to **Tools>SerialPort** and select the right serial port, the one arduino is attached to.

```
Blink | Arduino 0017
File Edit Sketch Tools Help
Blink $
int ledPin = 13; // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop()
{
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW); // set the LED off
  delay(1000); // wait for a second
}
```



Done compiling.

Press Compile button  
(to check for errors)



Upload

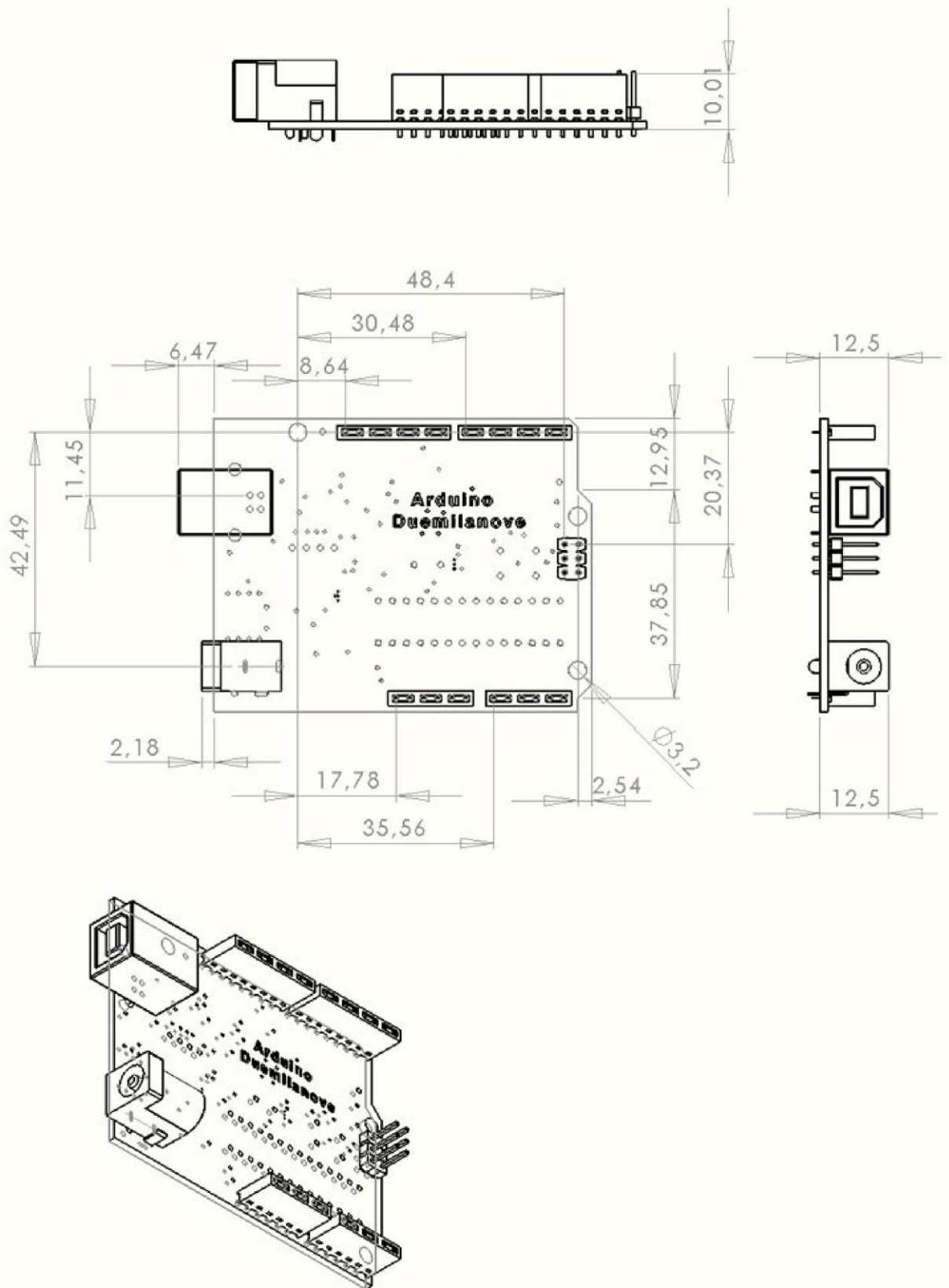


TX RX Flashing



Blinking Led!

# Dimensioned Drawing



# Terms & Conditions



## 1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino™ products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino™ products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino™ products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino™ products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

## 2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

## 3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

## 4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.



## Environmental Policies



The producer of Arduino™ has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.





**D-Link®**

Wireless AC750 Dual Band  
router Dir-809

# QUICK INSTALLATION GUIDE

КРАТКОЕ РУКОВОДСТВО ПО  
УСТАНОВКЕ GUÍA DE INSTALACIÓN  
RÁPIDA

GUIA DE INSTALAÇÃO RÁPIDA

快速安裝指南

PETUNJUK PEMASANGAN CEPAT



# CONTENIDO DEL EMPAQUE

## Contenido



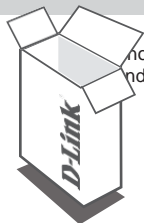
WIRELESS AC750 DUAL BAND ROUTER  
Dir-809



ADAPTADOR DE ALIMENTACIÓN  
A



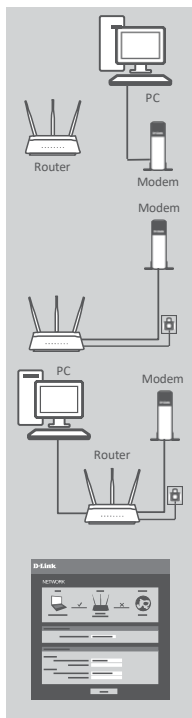
CABLE ETHERNET (CAT5 UTP)  
CONECTAR EL ROUTER AL PC DURANTE LA INSTALACIÓN



Si no encuentra alguno de los ítems mencionados, por favor contacte a su proveedor.

# INSTALACIÓN DEL PRODUCTO

## ConFiGurACiÓN BAsADA en PÁGiN A WeB



### Paso 1

Coloque el router cerca de su módem y el equipo que ya está en línea. Coloque el router abierta de su área de trabajo para una mejor cobertura inalámbrica.

### Paso 2

Conecte el cable Ethernet del módem que está conectado a su computador. Conecte el cable de alimentación incluido en el puerto de alimentación en la parte posterior del router y luego enchufe a una toma de corriente o protector contra rayos. Verifique que el indicador LED de alimentación está encendido.

### Paso 3

Conecte el otro extremo del cable Ethernet en el puerto con la etiqueta 1 en la parte posterior del router. Conecte el otro extremo de este cable al puerto Ethernet en su computador.

### Paso 4

Encienda el computador en el PC conectado y vaya a <http://192.168.0.1>. Seleccione su tipo de configuración de Internet (DHCP, PPPoE o IP estática), e introduzca los ajustes de Internet del proveedor de servicios de Internet (ISP). Haga clic en **Conectar** para aplicar la configuración.

# SOLUCIÓN DE PROBLEMAS

## Problemas De instalación Y Configuración

### 1. ¿CÓMO CONFIGURO MI ROUTER DIR-809 MANUALMENTE?

- Conecte su PC al router mediante un cable Ethernet.
- Abra el navegador web e introduzca la dirección '**192.168.0.1**'.
- El nombre de usuario predeterminado es Admin. La contraseña por defecto debe quedar en blanco (dejar este cuadro vacío).
- Si ha cambiado la contraseña y no lo recuerda, tendrá que reiniciar el dispositivo para establecer la contraseña por defecto.

### 2. ¿CÓMO PUEDO RESTABLECER MI ROUTER DIR-809 A LOS VALORES POR DEFECTO DE FÁBRICA?

- Asegúrese de que el dispositivo está encendido.
- Mantenga pulsado el botón de reset en la parte posterior del dispositivo durante 20 segundos. **Nota:** Al restablecer el producto a valores por defecto de fábrica, se borrará la configuración actual. Para volver a ajustar la configuración, inicie sesión como se indica en la pregunta 1, a continuación, ejecute el Asistente de configuración.

### 3. ¿CÓMO PUEDO AÑADIR UN NUEVO CLIENTE INALÁMBRICO O PC SI HE OLVIDADO MI NOMBRE DE RED INALÁMBRICA (SSID) O CLAVE DE CIFRADO INALÁMBRICO?

- Por cada PC que necesita conectar con el Router de forma inalámbrica, tendrá que asegurarse de que utiliza el nombre correcto de la red inalámbrica (SSID) y la clave de cifrado.
- Utilice la interfaz de usuario basada en web (como se describe en la pregunta 1) para comprobar o elegir la configuración inalámbrica.
- Asegúrese de anotar estos valores para que pueda ingresarlos en cada PC que desee conectar inalámbricamente. Va a encontrar un espacio especial en la parte posterior de este documento para escribir esta información importante.

### 4. ¿POR QUÉ NO PUEDO CONSEGUIR UNA CONEXIÓN A INTERNET?

- Para usuarios ADSL por favor contáctese con su ISP para asegurarse que el servicio está habilitado por su ISP y que su nombre de usuario ISP y contraseña son correctos.

Usted puede encontrar actualizaciones de softwares o firmwares y documentación para usuarios a través de nuestro sitio [www.dlinkla.com](http://www.dlinkla.com)

### **SOPORTE TÉCNICO PARA USUARIOS EN LATINO AMERICA**

Por favor revise el número telefónico del Call Center de su país en <http://www.dlinkla.com/soporte/call-center>

### **Soporte Técnico de D-Link a través de Internet**

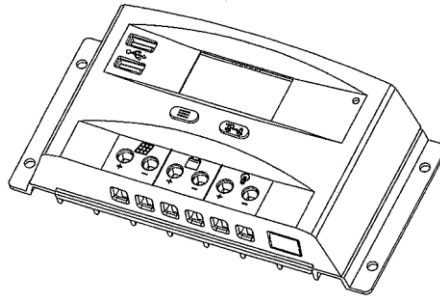
Horario de atención Soporte Técnico en [www.dlinkla.com](http://www.dlinkla.com)  
e-mail: [soporte@dlinkla.com](mailto:soporte@dlinkla.com) & [consultas@dlinkla.com](mailto:consultas@dlinkla.com)

Please ensure that channels 36/40/44/48 on the 5 GHz band are unlicensed and unrestricted in your country of residence before using them.

Any illegal usage of these channels is the responsibility of the user.

Por favor asegúrese de que los canales 36/40/44/48 en la banda de 5 GHz estén sin licencia y sin restricción de uso en su país de residencia antes de utilizarlos.

Cualquier uso ilegal de estos canales es responsabilidad del usuario.



Lea detenidamente este manual antes de utilizar el regulador.

## 1. Instrucciones de seguridad.

- 1.1. Mantenga su instalación lejos de cualquier dispositivo inflamable o explosivo, gases corrosivos, polvo, etc.
- 1.2. Proteja el regulador de la luz solar directa y de la lluvia.
- 1.3. Prevenga la ubicación del regulador para que no quede expuesta a líquidos o ambientes húmedos.
- 1.4. No desmonte el regulador.
- 1.5. No lo apoye sobre la batería. Su cuerpo metálico puede cortocircuitar los bornes de la misma.
- 1.6. No toque los terminales o el cuerpo metálico con las manos húmedas.

## 2. Introducción de producto.

### 2.1. Propósito

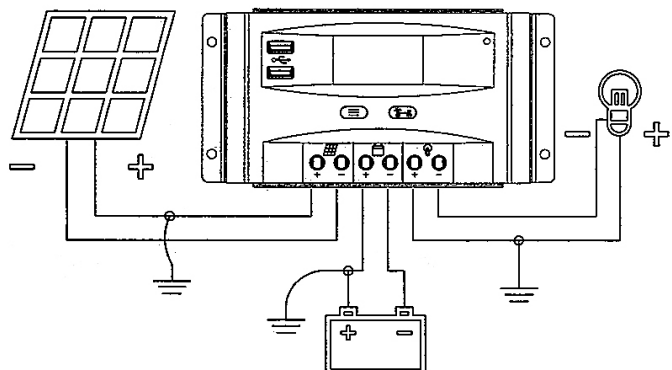
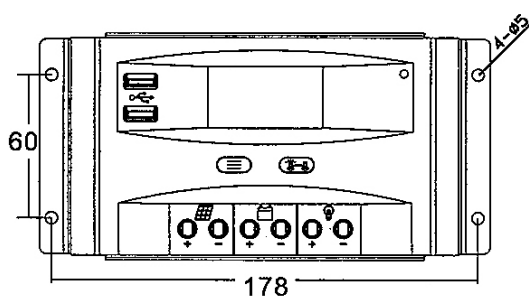
Esta gama de reguladores forma parte de una familia de controladores de carga de tipo PWM con avanzadas funciones de configuración y monitorización. Su diseño permite una instalación rápida y sencilla. Una carga y descarga optimizada prolonga la vida de las baterías de un modo considerable. Utilice siempre un controlador de carga para proteger la vida de sus baterías. Los parámetros de carga se muestran de manera explícita en la pantalla LCD.

### 2.2. Funciones

- 2.2.1. Carga multi-etapas PWM.
- 2.2.2. Ajustes predefinidos para 3 tecnologías de baterías de plomo.
- 2.2.3. Detección automática o manual del voltaje de trabajo de baterías.
- 2.2.4. Parámetros ajustables de carga y descarga.
- 2.2.5. Carga con compensación de temperatura.
- 2.2.6. Puerto opcional de comunicaciones (no incluido).
- 2.2.7. Protección para conexión inversa de panel y sobre-corriente de entrada.
- 2.2.8. Protección por bajo voltaje, exceso de voltaje, conexión inversa de batería y descarga inversa de batería.
- 2.2.9. Protección para sobre-corriente y cortocircuito en salida de consumo CC.
- 2.2.10. Protección interna por temperatura elevada.

## 3. Instalación

### 3.1. Tamaño y esquema unifilar.



3.2. Procedimiento de instalación.



Por favor, asegúrese de que la batería y el panel solar están desconectados de los cables que va a utilizar para conectar en el regulador. El contacto entre los cables positivo y negativo provocará un cortocircuito si éstos no están conectados a la batería o al panel. Una vez se conecten al regulador podrá conectar los otros extremos del cableado a baterías y a paneles en este mismo orden.

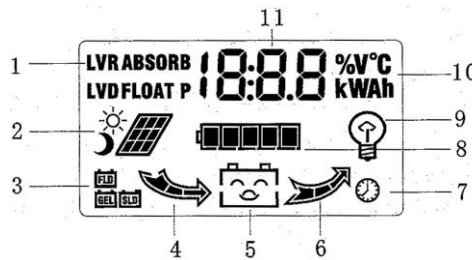
Deje un espacio disponible de al menos 15cm por cada lado para poder disipar el calor del propio regulador. Utilice cableado de un mínimo de 4mm<sup>2</sup> siendo recomendable utilizar cable solar de 6mm<sup>2</sup>.

- 3.2.1. Asegúrese que el lugar de instalación cumple con los mínimos requerimientos de seguridad.
- 3.2.2. Asegúrese de que el voltaje de baterías es el mismo que el de paneles y ambos son compatibles con este regulador.
- 3.2.3. Conecte primero la batería al regulador. Podrá ver que la pantalla LCD se enciende. Si no es así acuda al punto 5.2.
- 3.2.4. Conecte posteriormente el panel solar al regulador. Si hay producción el controlador empezará la carga de la batería inmediatamente y lo podrá observar sobre la propia pantalla.
- 3.2.5. Conecte las cargas en corriente continua en caso de disponer de ellas.
- 3.2.6. Si tiene que desconectar el regulador por cualquier motivo, debe hacerlo en orden inverso. Primero los consumos en corriente continua, posteriormente paneles solares y por último las baterías.

**4. Instrucciones de uso.**

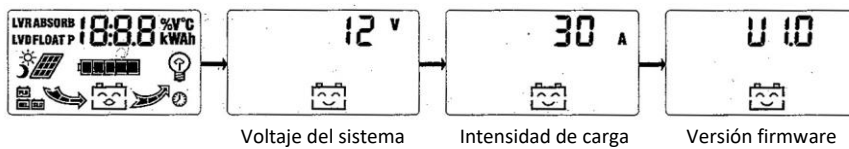
4.1. Símbolos en pantalla.

- 1-LVR, ABSORB, LVD, FLOAT, P (estado de carga de la batería)
- 2-Día y noche.
- 3-Tipo de batería seleccionada (FLD, GEL, SLD). Si no aparece, la batería seleccionada es USR.
- 4-Cargando desde panel.
- 5-Dibujo batería.
- 6-Descargando.
- 7-Temporizador.
- 8-Estado de carga estimado.
- 9- Estado de las cargas.
- 10-Unidad de medida.
- 11-Área de información numérica.



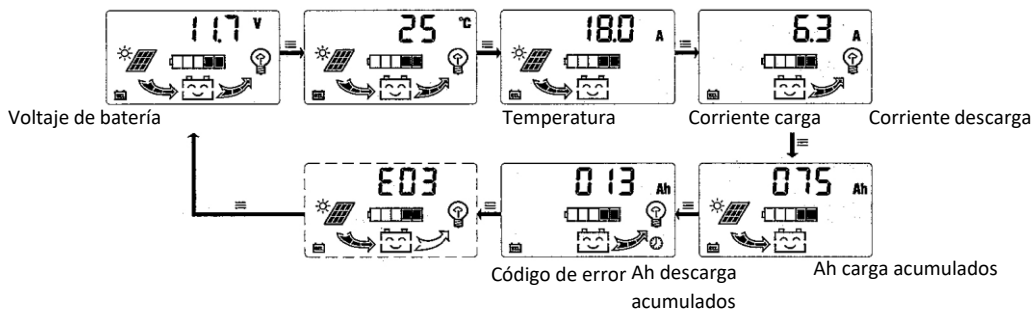
4.2. Menús pantalla LCD.

4.2.1. Secuencia de arranque.



- Se iluminan todos los ítems de la pantalla para poder ver que funciona correctamente.
- Voltaje de la batería detectado por el controlador.
- Intensidad de carga o de descarga máxima del sistema.

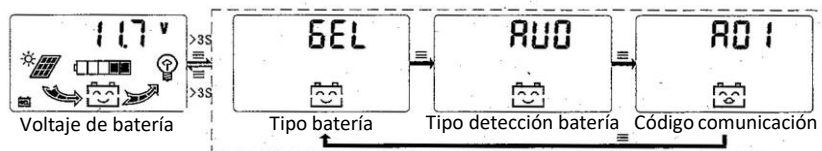
4.2.2. Secuencia principal de información en pantalla.





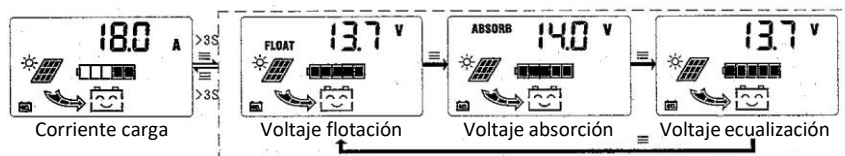
- Para ir avanzando entre las pantallas hay que pulsar el botón izquierdo. Una vez lleguemos a la última pantalla, volveremos a la primera de las opciones de visualización.
- En la pantalla “voltaje de batería” o “código de error” si pulsamos el botón derecho habilitaremos y deshabilitaremos la salida de consumo en corriente continua.
- En cualquiera de las pantallas de Ah acumulados, si pulsamos más de 3 segundos seguidos el botón izquierdo, el contador se reiniciará.
- En la pantalla “voltaje de batería” si pulsamos ambos botones simultáneamente de manera prolongada podremos resetear el regulador a sus ajustes de fábrica.

#### 4.2.3. Batería



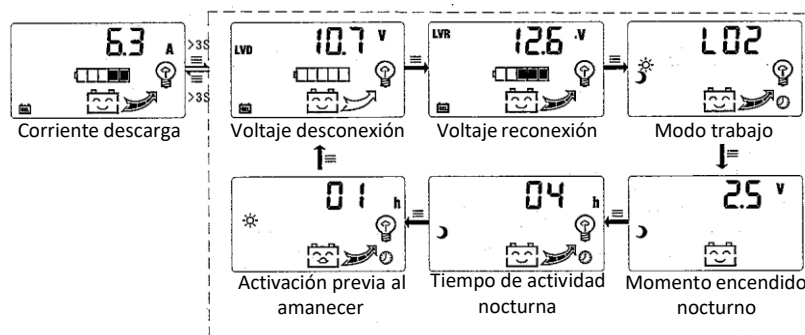
- En el menú principal estando en “voltaje de batería”, si pulsamos durante más de 3 segundos el botón izquierdo accederemos a éste submenú. Mediante pulsaciones cortas del botón izquierdo avanzaremos entre estas 3 opciones. Si hacemos una pulsación corta del botón derecho, podremos cambiar los valores de cada una de estas opciones. Para guardar los cambios haremos de nuevo una pulsación larga del botón izquierdo. Si no tocamos nada en 20 segundos, saldremos al menú principal sin guardar cambios.
- Se pueden establecer 4 tipos de batería para usar los voltajes predefinidos del fabricante (GEL, SLD, FLD y USR) Sólo en caso de elegir USR podremos cambiar los voltajes de cada etapa de carga a nuestro gusto.
- En la detección de batería puede estar en AUO (detección automática) o bien establecer nosotros manualmente el voltaje de trabajo de nuestra batería.
- El código de comunicaciones no tiene efecto puesto que el regulador no dispone de dicho puerto. El regulador se reinicia automáticamente cuando se cambia la tecnología de baterías o el voltaje del sistema.

#### 4.2.4. Ajustes de carga



- En el menú principal estando en “corriente carga”, si pulsamos durante más de 3 segundos el botón izquierdo accederemos a éste submenú. Mediante pulsaciones cortas del botón izquierdo avanzaremos entre estas 3 opciones. Si hacemos una pulsación corta del botón derecho, podremos cambiar los valores de cada una de estas opciones. Para guardar los cambios haremos de nuevo una pulsación larga del botón izquierdo. Si no tocamos nada en 20 segundos, saldremos al menú principal sin guardar cambios.
- Sólo se pueden realizar cambios en cada uno de estos apartados si hemos elegido el tipo de batería USR.

#### 4.2.5. Salida de consumo CC



- En el menú principal estando en “corriente carga”, si pulsamos durante más de 3 segundos el botón izquierdo accederemos a este submenú. Mediante pulsaciones cortas del botón izquierdo avanzaremos entre estas 3 opciones. Si hacemos una pulsación corta del botón derecho, podremos cambiar los valores de cada una de estas opciones. Para guardar los cambios haremos de nuevo una pulsación larga del botón izquierdo. Si no tocamos nada en 20 segundos, saldremos al menú principal sin guardar cambios.

Código de error	Causa	Solución
E01	Batería con exceso de descarga	Recargue la batería manualmente (sin el regulador)
E02	Exceso de consumo en cargas CC – se desconecta esta salida	Reducir consumos en la conexión CC y reconectar pulsando el botón izquierdo. O bien esperar 10 minutos para la
E03	Cortocircuito en cargas CC – se desconecta esta salida	Corregir el cortocircuito y reconectar pulsando el botón izquierdo. O bien esperar 10 minutos para la
E04	Fallo en la detección de la batería	Asegúrese de una conexión correcta y firme entre regulador y batería. Que la carga de la batería no sea excesivamente baja. Asegúrese de que si existe otro regulador sobre la misma batería, el voltaje del
E05	Carga de batería interrumpida debida a un exceso de temperatura	Permita que el regulador baje su temperatura y se reinicie automáticamente.
E06	Sobre-voltaje en paneles	Asegúrese de que el voltaje en circuito abierto no es demasiado elevado.
E07	Carga de batería interrumpida debido a un exceso de corriente en paneles	Compruebe la potencia de los paneles solares. Reduzca la potencia conectada y espere al

- Modos de trabajo de la salida de consumo CC.

Código	Modo de trabajo de consumo en CC
L00	Modo normal con activación manual.
L01	Control automático con activación tras anochecer y antes de amanecer.
L02	Control automático en iluminación.

Los diferentes modos de trabajo tienen configuraciones que se pueden ajustar en el submenú anterior.

## 5. Fallos

### 5.1. Códigos de error y solución a los mismos.

Fallo	Corrección
No hay señal ni puesta en marcha de la pantalla	Asegúrese de que la batería no está conectada al revés. Asegúrese de que la conexión entre batería y regulador es correcta. Asegúrese de que si tiene protecciones entre ambos componentes estén en la posición correcta (desconectador, fusible, etc.)
No hay carga hacia la batería	Asegúrese de que el panel no está conectado al revés. Asegúrese de que el cableado entre panel y batería no está interrumpido y que llega tensión a los extremos que están en el regulador.
La salida de consumo no funciona	Asegúrese de que la conexión en cargas CC no está conectada al revés. Asegúrese de que el controlador no tiene activada la protección por sobrecarga, cortocircuito, sobre-voltaje o voltaje bajo.
La salida de consumo no se activa cuando la hemos programado	Asegúrese de que el modo de salida de consumo está correctamente configurado. Asegúrese que el voltaje de batería no es excesivamente bajo.
La salida de consumo no se activa cuando anochece con el modo automático nocturno	Asegúrese de que el modo de salida de consumo está correctamente configurado. Asegúrese de que el panel solar no recibe ningún tipo de iluminación de otras fuentes.

## 6. Post-Venta

### 6.1. Garantía.

La garantía del fabricante cubre cualquier daño del regulador en los 2 primeros años de uso siempre que su mal funcionamiento sea ajeno al usuario. Cualquier manipulación del aparato conllevará la anulación de la garantía. Todos los fallos derivados de no seguir las instrucciones del presente manual conllevarán la anulación de la garantía. Para cualquier duda consulte con su punto de venta o comercial.

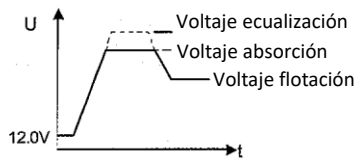
## 7.DATOS TÉCNICOS

Modelo		10A 12/24V	20A 12/24V	30A 12/24V
Entrada	Voltaje FV	≤50V		
	Intensidad nominal	10A	20A	30A
Salida	Voltaje sistema	12V/24V Auto		
	Desconexión por alto voltaje (HVD)	16.00V x1 / x2		
	Intensidad descarga nominal	10A	20A	30A
	Autoconsumo	≤13mA		
	Caída de tensión circuito carga	≤0.21V		≤0.24V
	Caída de tensión circuito descarga	≤0.12V		≤0.1V
	Modo de carga	PWM Multi-etapas (carga, absorción, flotación,		
	Voltaje Carga Flotación	13.8V (13V~15V) x1 / x2		
	Voltaje Carga Absorción	2 horas duración	14.4V (13V~15V) x1 / x2	
	Voltaje Carga Ecuilización		14.6V (13V~15.5V) x1 / x2	
	Protección Bajo Voltaje (LVD)	10.8V (10V~14V) x1 / x2		
	Reconexión Bajo Voltaje (LVR)	12.6V (10V~14V) x1 / x2		
	Salida USB	5V 1A		
Características físicas	Sección cableado	6mm <sup>2</sup>	10mm <sup>2</sup>	16mm <sup>2</sup>
	Temperatura trabajo	-20 ~ +50°C		
	Tamaño (L x W x H)	188 x 95 x 46.5mm		
	Peso neto	355g		

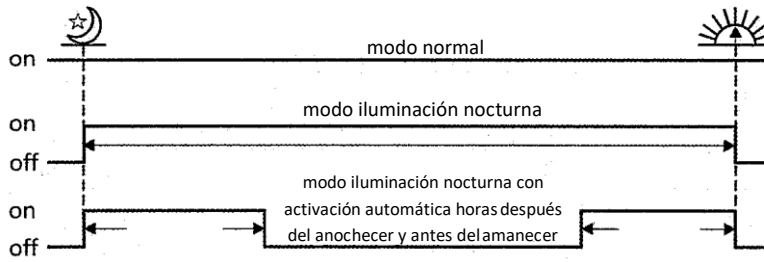
Tipo batería	Voltaje de flotación	Voltaje de absorción	Tiempo de absorción (h)	Voltaje de ecualización	Tiempo de ecualización (h)	Intervalo entre ecualizaciones (días)
GEL	13.8	14.2	2	-	-	-
Sellada (SLD)	13.8	14.4	2	14.6	2	28
Plomo abierto (FLD)	13.8	14.6	2	14.8	2	28
Personalizada (USR)	13.8	14.4	2	14.6	2	28

En el anterior cuadro se ve los tipos de batería.

### Modo de carga



### Modo de consumo



# ON TAT INDUSTRIAL COMPANY

## SPECIFICATION FOR LCD MODULE

Customer : \_\_\_\_\_

Product Model:           KD50G<sup>2</sup>1-40NT-A1          

Sample code: \_\_\_\_\_

Designed by	Checkedby	Approved by

### Final Approval by Customer

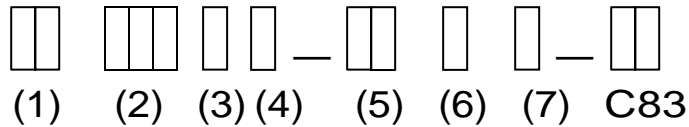
<input type="checkbox"/>	LCM Machinery OK CheckedBy _____	<input type="checkbox"/>	LCM OK
<input type="checkbox"/>	LCM Display OK CheckedBy _____	<input type="checkbox"/>	NG , Problem survey:  Approved By _____


☒ The specification of "TBD" should refer to the measured value of sample . If there is difference between the design specification and measured value, we naturally shall negotiate and agree to solution with customer.





1 Numbering System



No	Definition	Specifica
(1)	TFT LCM Productor No.	On Tat Industrial Company
(2)	Display monitor opposite angle	Unit :mm or mmm (size <10 inch: takes two integers ; size >=10 inch: takes three integers )
(3)	Productor Types	D---- Digital photo frame / DVD G ----GPS M---- MP P ----Mobil-Phone 
(4)	Productor Development Series No.	By two figures characters expression from 01 to 99
(5)	Interface PIN Number	By two figures characters expression from 01 to 99
(6)	With Touch Panel Or Not	T----With T/P = N----Without T/P
(7)	LCD Type	A----AUO = M----CMO = C----CPT = P----PVI = L----LG = W----Wintek = H----HSD = T----Tianma =
(8)	Productor Development edition No.	By The English litters : A 1~ Z9



## 2 Scope

This specification applies to the TFT LCD module which is designed and manufactured by LCM Factory of On Tat Industrial Company

## 3 Normative Reference

GB/T4619-1996 Liquid Crystal Display Test Method

GB/T2424 Basic environmental Testing Procedures for Electric and Electronic Products.

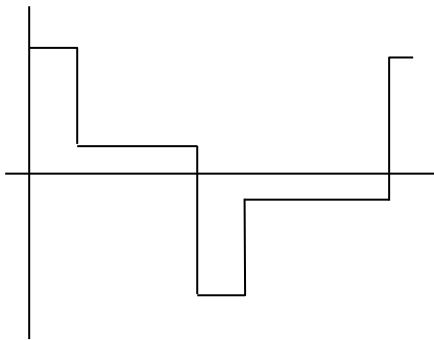
GB/T2423 Basic Testing Procedures for Electric and Electronic Products

IEC61747-1 SIXTH PART GB2828`2829-87 National Standard of PRC

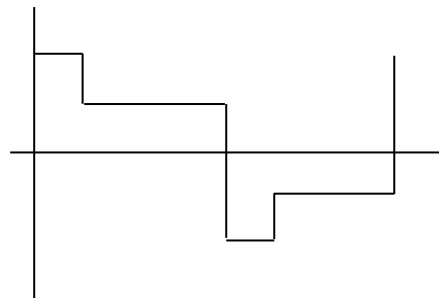
## 4 Definitions

### 4.1 Definitions of Vop

The definitions of threshold voltage Vth,1 Vth2 the following typical waveforms are applied on liquid crystal by the method of equalized voltage for each duty and bias.



[ selected waveform ]



[ non-selected waveform ]

Vth1: The voltage which the brightness of segment indicates 50% of saturated value on the conditions of selected waveform

( $f_r=80\text{Hz}$ ,  $\theta = 10^\circ$  or  $=270^\circ$  at 25?) )

Vth2: The voltage which the brightness of segment indicates 50% of saturated value on the conditions of non-selected waveform

( $f_r=80\text{Hz}$ ,  $\theta = 10^\circ$  or  $=270^\circ$  at 25?) )

⑤ Vop:  $(V_{th1}(50\%)+V_{th2}(50\%))/2$  ( $f_r=80\text{Hz}$ ,  $\theta = 10^\circ$  or  $=270^\circ$  at 25?) )

### 4.2 Definition of Response Time Tr, Td

Tr: The time required which the brightness of segment becomes 10% from 100% when waveform is switched to selected one from non-selected one. ( $f_r=80\text{Hz}$ ,  $\theta = 10^\circ$  or  $=270^\circ$  at 25?) )

Td: The time required which the brightness of segment becomes 90% from 10% when waveform is switched to selected one from selected one. ( $f_r=80\text{Hz}$ ,  $\theta = 10^\circ$  or  $=270^\circ$  at 25?) )

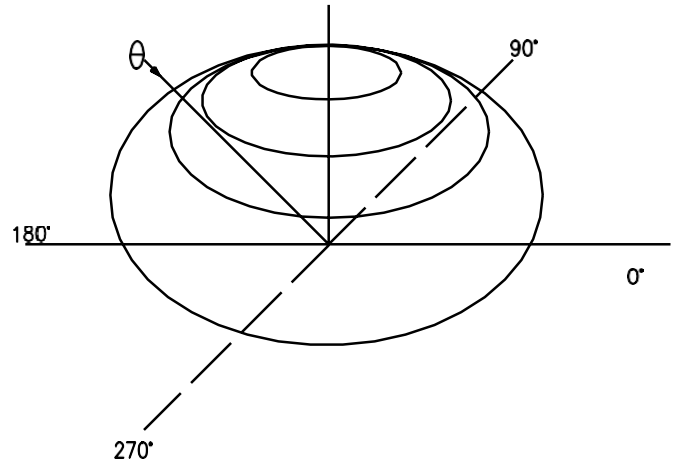
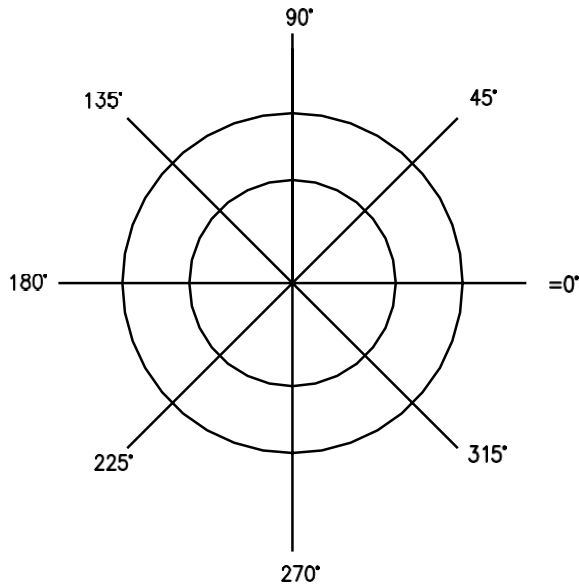
4.3 Definition of Contrast Ratio Cr

$$Cr = A/B$$

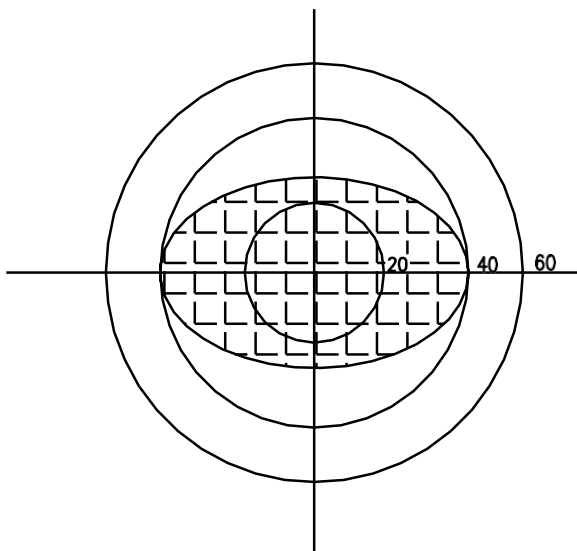
A: Segments brightness in case of non-selected waveform

B: Segments brightness in case of selected waveform

4.4 Definition of Angle and Viewing Range



Angular Graph: Contrast Ratio



Such as:  
Viewing Angle Range:  
80(Cr>2) Horizontal  
70(Cr>2) Vertical

## 5 Technology Specifications

### 5.1 Feature

This single-display module is suitable for use in Multimedia Player products. The LCD adopts one backlight with High brightness 12-lamps white LED.

- 1) Construction: 5.0" a - i color TFT-LCD ,White LED backlight.
- 2) LCD:
  - 2.1 Amorphous-TFT 5.0 inch display, transmissive, normally white type.
  - 2.2 800(RGB)×480 dots Matrix.
  - 2.3 Narrow-contact ledge technique.
  - 2.4 LCD Driver IC: HX8264D × 1 and Hx8664B × 1.
- 3) Low cross talk by frame rate modulation.
- 4) RGB interface.
- 5) Video signal interface: Parallel RGB or serial.

### 5.2 Mechanical Specifications

Item	Specifications	Unit
Dimensional outline	120.70(W) ×75.8(H)×3.10 Max(T)	mm
Active area	108.0(W) ×64.8 (H)	mm
Pixel size	135(W) ×135(H)	um
Resolution	800(RGB) × 480	pixel
Luminance for LCM	200	cd/m <sup>2</sup>

### 5.3 Absolute Max. Rating

Item	Symbol	Value			Unit	Remark
		Min	typ	Max		
Logic power supply	DVDD	3.0	3.3	3.6	V	
Driver power supply	AVDD	11	12	13	V	
Input high voltage	V <sub>IH</sub>	0.7DVDD	-	DVDD	V	
Input low voltage	V <sub>IL</sub>	0	-	0.3DVDD	V	
Operating temperature	T <sub>OPR</sub>	-20		+70	?	
Storage temperature	T <sub>STG</sub>	-30		+80	?	

### 5.4 Electrical Characteristics (VSS=0V, Ta=-20 to 70°C)

Parameter	Symbol	Applicable pin	Condition	Spec.			Unit
				Min.	Typ.	Max.	
Input H voltage	V <sub>IH</sub>	All input pins <sup>(2)</sup>	-	0.7VDD	-	VDD	V
Input L voltage	V <sub>IL</sub>	All input pins <sup>(2)</sup>	-	VSS	-	0.3VDD	
Input H voltage	V <sub>IH</sub>	/XAO		0.8VDD	-	VDD	
Input L voltage	V <sub>IL</sub>	/XAO		VSS	-	0.2VDD	
Output H voltage	V <sub>OH</sub>	STV1,2	I <sub>OH</sub> =40μA	VDD-0.4	-	VDD	
Output L voltage	V <sub>OL</sub>	STV1,2	I <sub>OL</sub> =40μA	VSS	-	VSS+0.4	
Output H resistance	R <sub>OH</sub>	OUT0~ OUT961	V <sub>OUT</sub> = VGH-0.5V	-	-	1000	Ω
Output L resistance	R <sub>OL</sub>	OUT0~ OUT961	V <sub>OUT</sub> = VEE+0.5V	-	-	1000	Ω
Input leakage current	I <sub>IN</sub>	Note <sup>(2)</sup>	-	-5.0	-	+5.0	μA
Pull high resistance	R <sub>PU</sub>	/XAO, MODE, SEL	VDD=3.3V, TA=25C	70	160	400	kΩ
VGH Power consumption	I <sub>VGH</sub>	-	Note <sup>(1)</sup>	-	-	200	μA
VDD Power consumption	I <sub>VDD</sub>	-		-	-	200	

Note: (1) Power consumption in the following condition:

Output no load, VGH=20V, VEE=-8V, VDD=3.0V, V<sub>IH</sub>=VDD, V<sub>IL</sub>=VSS, F<sub>CPV</sub>=50KHz, OE=V<sub>IL</sub>, /XAO=V<sub>IH</sub>.

(2) All input pins except /XAO, MODE, SEL

OKB/S

Parameter	Symbol	Condition	Spec.			Unit
			Min.	Typ.	Max.	
CPV period	t <sub>CPV</sub>	-	5	-	-	μs
CPV pulse width	t <sub>CPVH</sub> , t <sub>CPVL</sub>	50% duty cycle	2.5	-	-	
OE pulse width	t <sub>WOE</sub>	-	1	-	-	
/XAO pulse width	t <sub>WXAO</sub>	-	30	-	-	
Data setup time	t <sub>SU</sub>	-	0.3	-	-	
Data hold time	t <sub>HD</sub>	-	0.3	-	-	
CPV to output delay time	t <sub>PD1</sub>	CL=200pF	-	-	0.9	
Start pulse output delay time	t <sub>PD2</sub>	Loading=20pF	-	-	0.5	
OE to output delay time	t <sub>PD3</sub>	CL=200pF	-	-	0.9	
/XAO to output delay time	t <sub>PD4</sub>	CL=200pF	-	-	50	

Note: The measurement point for all of above signals is at 50% of input/output amplitude under Typical condition.

5.5 Optical specifications

Ta=25°C

Item	Symbol	Condition	Min	Typ.	Max	Unit	Remark	
View Angle	$\theta T$	$CR \geq 10$	40	50	-	Degree	Note 2	
	$\theta B$		60	70	-			
	$\theta L$		60	70	-			
	$\theta R$		60	70	-			
Contrast Ratio	CR	$\theta=0^\circ$	400	500	-	-	Note1	
Response Time	$T_{ON}$	25°C	-	20	25	ms	Note1	
	$T_{OFF}$						Note4	
Color Filter Chromaticity	White	C-light	x	0.260	0.310	0.360	-	Just CF, without polarizer
			y	0.280	0.330	0.380		
	Red		x	0.540	0.590	0.640		
			y	0.300	0.350	0.400		
	Green		x	0.298	0.348	0.398		
			y	0.520	0.570	0.620		
	Blue		x	0.095	0.145	0.195		
			y	0.060	0.110	0.160		
NTSC	-	-	45	50	-	%	Note 5	
Transmittance	L	-	3.8	4.1	-	%	Note1	

Test Conditions:

1. The ambient temperature is 25°C.
2. The test systems refer to Note 1 and Note 2.
3. Viewing angle, contrast ratio and transmittance are test the panel include EWW polarizer (NWF-LNSW) with LED backlight.
4. The value of transmittance in this spec is a reference which will be affected by polarizer finally.

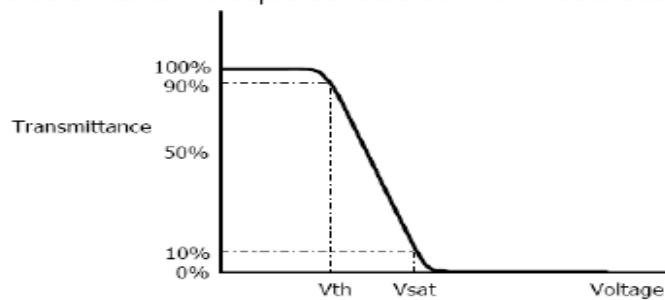
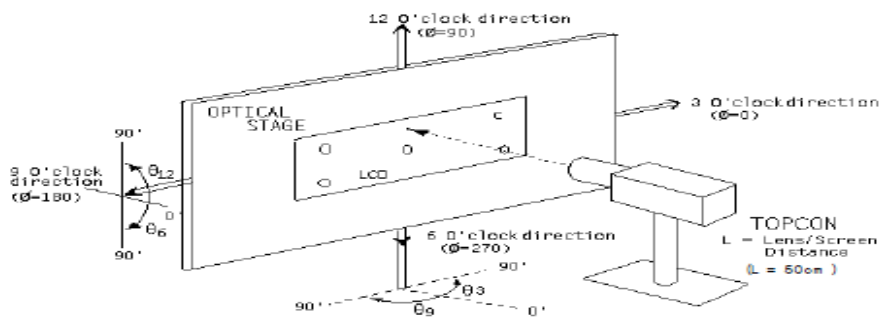


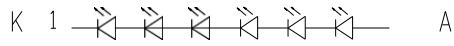
Figure 2. Measurement Set Up



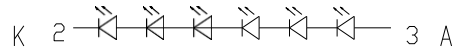
## 5.6 LED back light specification (12 White Chips)

Item	Symbol	Condition	Min	Typ	Max	Unit
Forward Voltage	V <sub>f</sub>	I <sub>f</sub> =40mA	18	19.2	20.4	V
Uniformity (with L/G)	A B <sub>0</sub>	I <sub>f</sub> =40mA	80	-	-	%
Luminance for BL	L <sub>v</sub>	I <sub>f</sub> =40mA	4800	-	-	cd/m <sup>2</sup>

## LED CIRCUIT



4

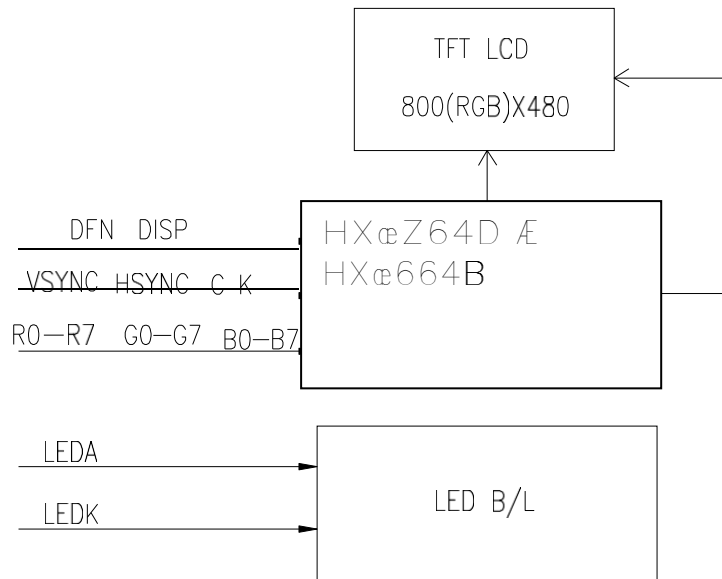


## 5.7 Interface PinConnections

PIN NO.	Symbol	Description
1	LED-K	LED backlight(Cathode)
2	LED-A	LED backlight(anode)
3	GND	Ground
4	VDD	Power supply (Digital +3.0V)
5-12	R0-R7	Red Data
13-20	G0-G7	Green Data
21-28	B0-B7	Blue Data
29	GND	Ground
30	CLK	Clock <input type="checkbox"/> <u>Latch data at clk falling edge</u> <sup>3</sup>
31	DISP	Display on/off
32	HSYNC	Horizontal sync input in RGB mode (short to GND if not used)
33	VSYNC	Vertical sync input in RGB mode (short to GND if not used)
34	DEN	Data Enable
35	NC	NC
36	GND	Ground
37	NC (XR)	NC(Touch panel coordinate in the right side of envisage drawing)
38	NC (YD)	NC(Touch panel coordinate in the down side of envisage drawing)
39	NC (XL)	NC(Touch panel coordinate in the left side of envisage drawing)
40	NC (YU)	NC(Touch panel coordinate in the up side of envisage drawing)

## 6 Signal timing diagram and Circuit block diagram

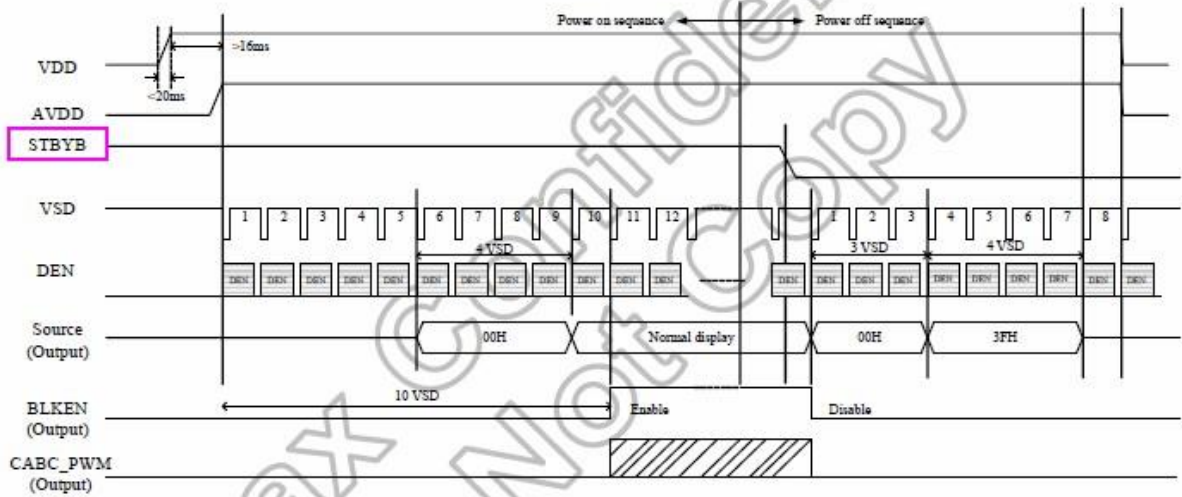
### 6.1 Circuit block diagram



## 6.2 Signal Timing Diagram

### 6.2.1 Power ON/OFF Sequence

HX8264-D02 has a power on/off sequence control function. In order to prevent IC from power on reset fail, the rising time ( $T_{POR}$ ) of the digital power supply VDD should be maintained within the given specifications. Please refer to “AC Characteristics” for more detail on timing.

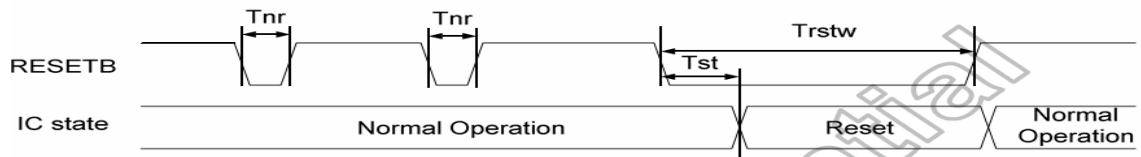


### 6.2.2 Reset timing

The reset input must be held for at least 1ms after power is stable.

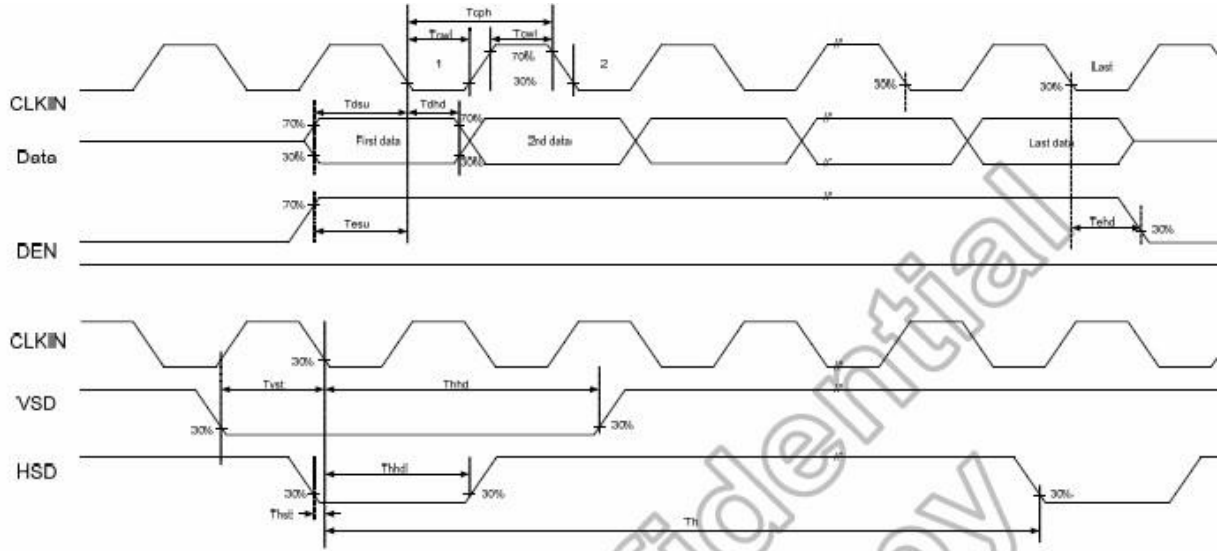
#### Hardware reset timing

PARAMETER	Symbol	Spec.			Unit
		Min.	Typ.	Max.	
RESETB low pulse width	$T_{rstw}$	10	-	-	$\mu s$
Negative noise pulse width	$T_{nr}$	-	-	2	$\mu s$
Reset start time	$T_{st}$	2	-	-	$\mu s$



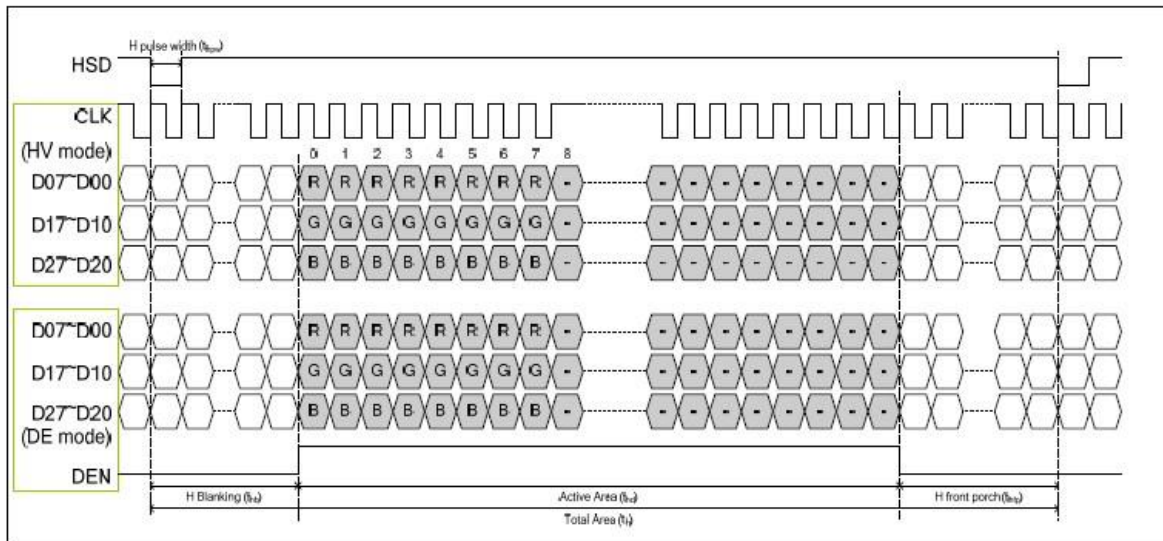


6.2.3 Timing Diagram of interface Signal

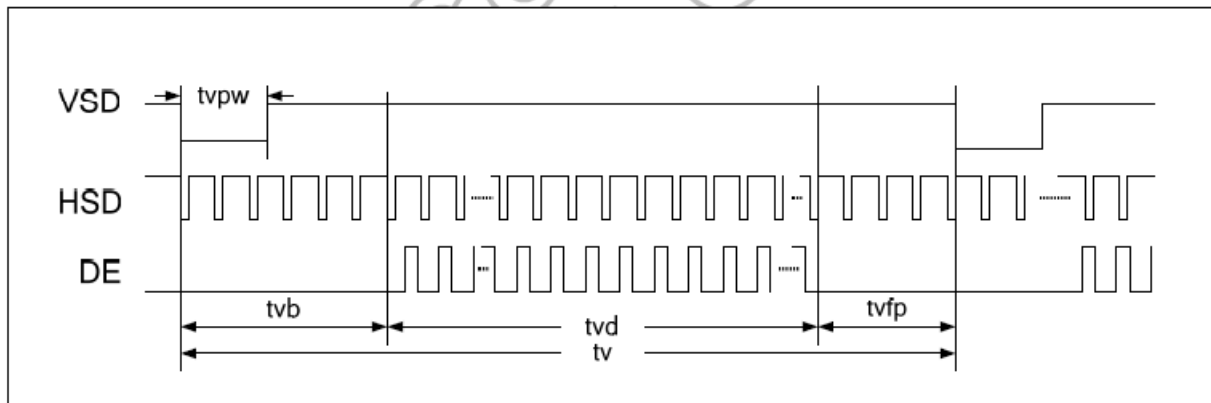


Parameter	Symbol	Spec.			Unit
		Min.	Typ.	Max.	
HS setup time	$T_{hst}$	8	-	-	ns
HS hold time	$T_{hhd}$	8	-	-	ns
VS setup time	$T_{vst}$	8	-	-	ns
VS hold time	$T_{vhd}$	8	-	-	ns
Data setup time	$T_{dsu}$	8	-	-	ns
Data hold time	$T_{dhd}$	8	-	-	ns
DE setup time	$T_{esu}$	8	-	-	ns
DE hold time	$T_{ehd}$	8	-	-	ns
VDD Power On Slew rate	$T_{POR}$	-	-	20	ms
RSTB pulse width	$T_{Rst}$	10	-	-	us
CLKIN cycle time	$T_{cph}$	20	-	-	ns
CLKIN pulse duty	$T_{cwh}$	40	50	60	%
Output stable time	$T_{sst}$	-	-	6	us

● Horizontal timing



- Vertical timing



- Horizontal timing

Parameter	Symbol	Spec.			Unit
		Min.	Typ.	Max.	
Horizontal Display Area	thd		800		DCLK
DCLK frequency	fclk	-	30	50	MHz
One Horizontal Line	th	889	928	1143	DCLK
HS pulse width	thpw	1	48	255	DCLK
HS Back Porch (Blanking)	thb		88		DCLK
HS Front Porch	thfp	1	40	255	DCLK
DE mode Blanking	th-thd	85	128	512	DCLK

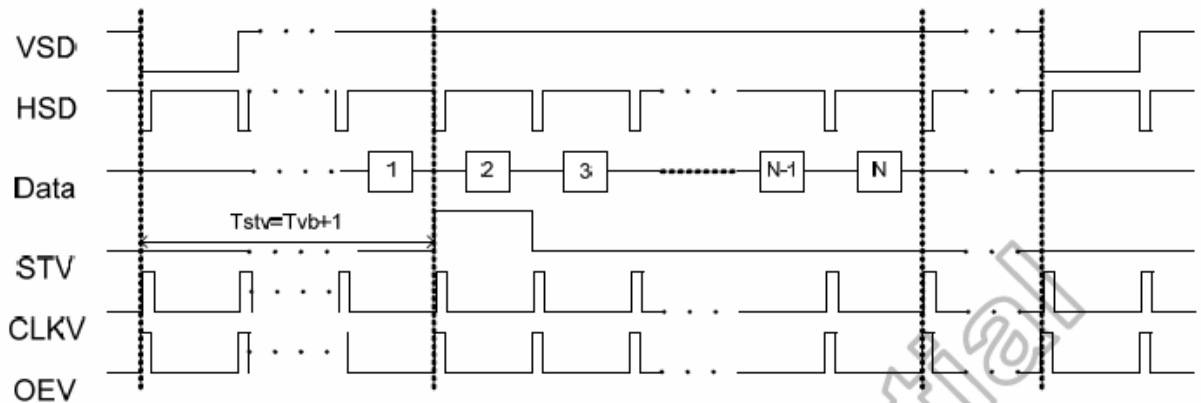
- Vertical timing

Parameter	Symbol	Spec.			Unit
		Min.	Typ.	Max.	
Vertical Display Area	tv_d		480		$T_H$
VS period time	tv	513	525	767	$T_H$
VS pulse width	tv <sub>pw</sub>	3	3	255	$T_H$
VS Back Porch (Blanking)	tv <sub>b</sub>		32		$T_H$
VS Front Porch	tv <sub>fp</sub>	1	13	255	$T_H$
DE mode Blanking	tv-tv <sub>d</sub>	4	45	255	$T_H$

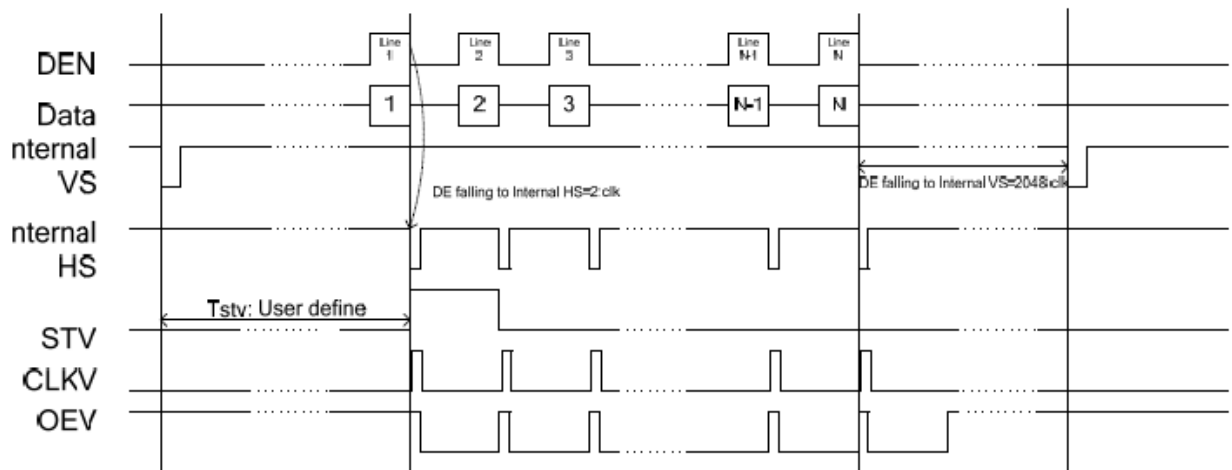
### 6.2.4 Driver Timing Control

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
CLKIN Frequency	Fclk	-	40	50	MHz	VDD=3.0V~3.6V
CLKIN Cycle Time	Tclk	20	25	-	ns	-
CLKIN Pulse Duty	Tcwh	40	50	60	%	Tclk
Time from HSD to Source Output	Thso		64		CLKIN	-
Time from HSD to LD	Thld		64		CLKIN	-
Time from HSD to STV	Thstv		2		CLKIN	-
Time from HSD to CKV	Thckv		20		CLKIN	-
Time from HSD to OEV	Thoev		4		CLKIN	-
LD Pulse Width	Twld		10		CLKIN	-
CKV Pulse Width	Twckv		66		CLKIN	-
OEV Pulse Width	Twoev		74		CLKIN	-

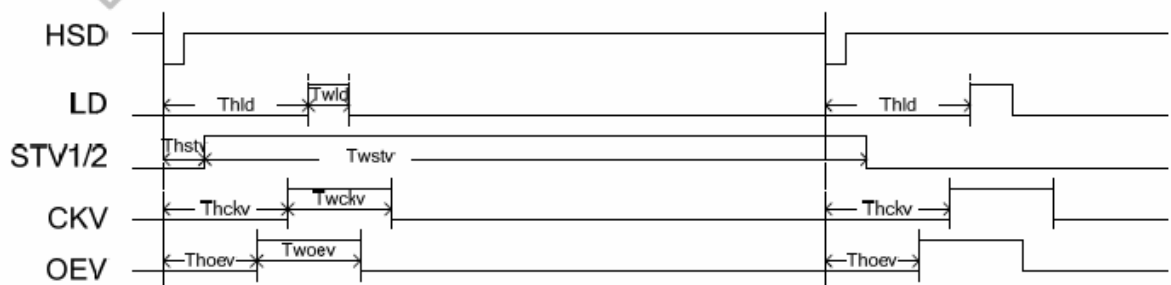
#### Vertical timing diagram HV (Cascade)



#### Vertical timing diagram DE (Cascade)



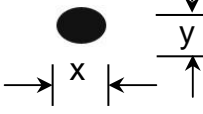
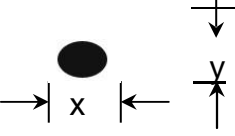
#### Gate output timing diagram (Cascade)

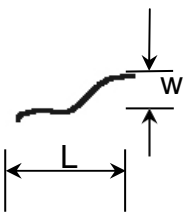
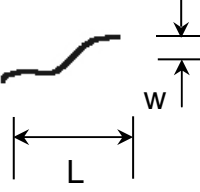


## 7 Reliability Test Conditions And Methods

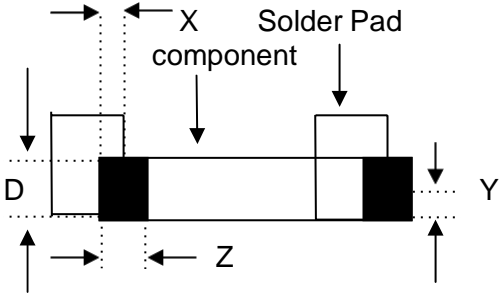
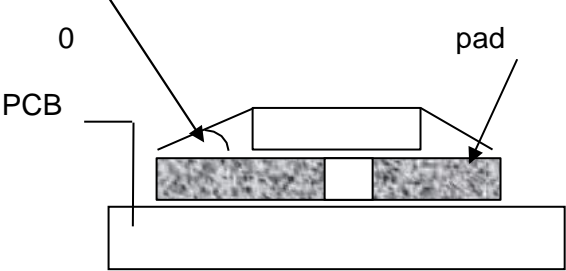
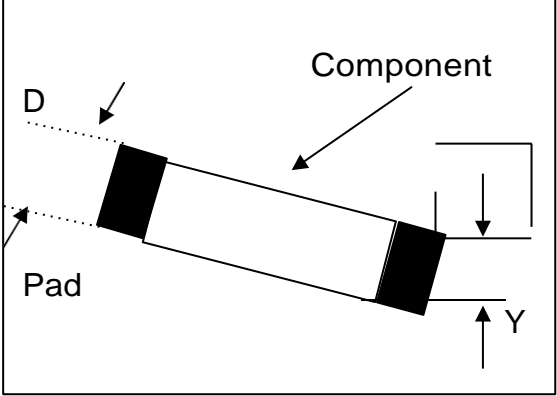
NO	Item	Condition	Method
1	High / Low Temperature Storage	80° /-30° 120hrs	Check and record every 48Hrs
2	High / Low Temperature Life	70° /-20° 120hrs (operating mode)	Check and record every 48Hrs
3	High Temperature , High Humidity Operating	60° ,90% RH, 96Hrs	Check and record every 48hrs
4	Thermal Shock	-30° (30Min) → 25° (5Min) → 80° (30Min) (conversion time, : 5 sec ) 20 cycles	Each 10 cycles end , check
5	Vibration	10Hz~55Hz~10Hz Amplitude: 1.5mm 2hrs for each direction(X,Y,Z)	Each direction end, Check the Appearance and Electrical Characteristics
6	Static Electricity	Gap mood: ±1KV~±8KV (10 times air discharge with positive/negative voltage voltage gap : 1kv) Touch mood: ±1KV~±4KV	Each discharge end, Check the Electrical Characteristics
7	Curve	60 Thousand times, 40 times/min 150° ( according to die if exist)	Check and record every 2~4 thousand times
8	Slump	Free faller movement for each side, cording, angle (75cm High, 6 sides, 2 angle, 2 cording)	End

### 8 Inspection standard

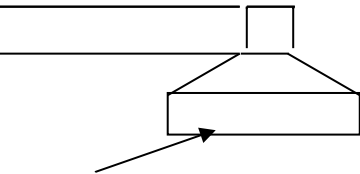
No	Item	Criterion																			
01	Outline Dimension	In accord with drawing																			
02	Position-finding Dimension Assemble Dimension	In accord with drawing																			
03	LCD black spots, white spots (Round type)	<p>Round type: non display</p> <p>3.1 Small area LCD</p> <p style="text-align: right;">Unit : mm</p>  <table border="1" data-bbox="775 752 1300 1061"> <thead> <tr> <th>Dimension</th> <th>Qualified Quantity</th> </tr> </thead> <tbody> <tr> <td>D" 0.1</td> <td>Ignore</td> </tr> <tr> <td>0.1 &lt; D" 0.15</td> <td>2</td> </tr> <tr> <td>D &gt; 0.15</td> <td>0</td> </tr> </tbody> </table> <p>3.2 Large area LCD</p>  <table border="1" data-bbox="767 1245 1300 1630"> <thead> <tr> <th>Dimension</th> <th>Qualified Quantity</th> </tr> </thead> <tbody> <tr> <td>D" 0.1</td> <td>Ignore</td> </tr> <tr> <td>0.1 &lt; D" 0.15</td> <td>2</td> </tr> <tr> <td>0.15 &lt; D" 0.20</td> <td>1</td> </tr> <tr> <td>D &gt; 0.20</td> <td>0</td> </tr> </tbody> </table> <p>C-STN : if D &gt; 0.1 , unqualified</p>	Dimension	Qualified Quantity	D" 0.1	Ignore	0.1 < D" 0.15	2	D > 0.15	0	Dimension	Qualified Quantity	D" 0.1	Ignore	0.1 < D" 0.15	2	0.15 < D" 0.20	1	D > 0.20	0	
Dimension	Qualified Quantity																				
D" 0.1	Ignore																				
0.1 < D" 0.15	2																				
D > 0.15	0																				
Dimension	Qualified Quantity																				
D" 0.1	Ignore																				
0.1 < D" 0.15	2																				
0.15 < D" 0.20	1																				
D > 0.20	0																				

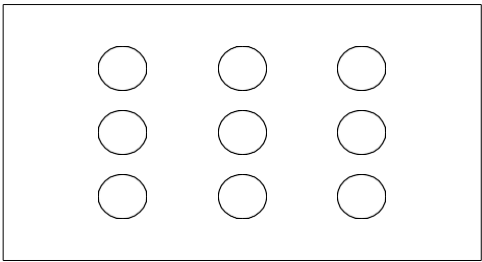
04	LCD Black spots, white spots (Line Style)	Unit : mm		4.1	Small	area	LCD														
			<table border="1"> <thead> <tr> <th>Length</th> <th>Width</th> <th>Qualified Quantity</th> </tr> </thead> <tbody> <tr> <td>-</td> <td>" 0.015</td> <td>Ignore</td> </tr> <tr> <td>" 1.0</td> <td rowspan="2">0.015 &lt; W &lt; 0.025</td> <td>2</td> </tr> <tr> <td>" 2.0</td> <td>1</td> </tr> <tr> <td>" 1.0</td> <td>0.025 &lt; W &lt; 0.05</td> <td>1</td> </tr> <tr> <td>-</td> <td>D &gt; 0.05</td> <td>According to circle</td> </tr> </tbody> </table>	Length	Width	Qualified Quantity	-	" 0.015	Ignore	" 1.0	0.015 < W < 0.025	2	" 2.0	1	" 1.0	0.025 < W < 0.05	1	-	D > 0.05	According to circle	
Length	Width	Qualified Quantity																			
-	" 0.015	Ignore																			
" 1.0	0.015 < W < 0.025	2																			
" 2.0		1																			
" 1.0	0.025 < W < 0.05	1																			
-	D > 0.05	According to circle																			
			4.2	Large area	LCD																
		<table border="1"> <thead> <tr> <th>Length</th> <th>Width</th> <th>Qualified Quantity</th> </tr> </thead> <tbody> <tr> <td>-</td> <td>" 0.015</td> <td>Ignore</td> </tr> <tr> <td>" L &gt; 2.0</td> <td>0.015 &lt; W &lt; 0.025</td> <td>2</td> </tr> <tr> <td>" 1.0</td> <td>0.025 &lt; W &lt; 0.05</td> <td>1</td> </tr> <tr> <td>-</td> <td>D &gt; 0.05</td> <td>According to circle</td> </tr> </tbody> </table>	Length	Width	Qualified Quantity	-	" 0.015	Ignore	" L > 2.0	0.015 < W < 0.025	2	" 1.0	0.025 < W < 0.05	1	-	D > 0.05	According to circle				
Length	Width	Qualified Quantity																			
-	" 0.015	Ignore																			
" L > 2.0	0.015 < W < 0.025	2																			
" 1.0	0.025 < W < 0.05	1																			
-	D > 0.05	According to circle																			
		CSTN : If W " 0.015 , unqualified Ignore beyond viewing area																			
05	LCD Scratch , Threadlike Fiber	Same to NO.3 circle sightline and surface of LCD is vertical (2)Same to NO.3 line style																			
06	POL	It is not admissible that POL is beyond the edge of glass, else, unqualified. It is essential that POL is over the 50 percent of width of frame , else ,unqualified. According to the drawing in case of special definition.																			
07	IC/FPC Bonding	Scratch	Reject																		

		Intensity Of Adhesion	If lower than specification, reject	
		Gold Fold Twist	Reject	
07	IC/FPC Bonding	Silicon	According to outline, no gold outside, seal can not be higher than LCD	
		FPC Gold Sever	Reject	
08	SMT	Lack of Component , Polarity Inverse	If exist, reject	
		Leak Solder , Virtual Solder	If exist, reject	
		Short Circuit In Solder Point	If exist, reject	
		Tin Ball	If exist, reject	
		Tin Acumination	If visual, reject	
		Height Solder Point	If higher 0.5mm than component. reject	
		Height of component	Either side higher 0.5mm than component, reject	

		Component Shift	 <p> <math>X &lt; 3/4Z</math> reject  <math>y &gt; 1/3D</math> reject         </p>
08	SMT	Few Tin	 <p>If <math>0 \leq \theta &lt; 20^\circ</math> reject</p>
		Component Deflection	 <p>If <math>Y &gt; 1/3D</math> reject</p>
		Component Carcass Sideways	Reject



		Component Carcass Sideways	If exist with visual inspection , reject	
		Lot Tin	A: Tin accrete the solder side completely , hollowly ,Ok B: Tin accrete the solder side completely , full circle arc , ok C: Jointing include whole solder side, height of tin>50 percent of height of component, reject	
		Few Tin	A: Tin accrete the solder side completely , hollowly ,Ok B: height of tin > 1/3 of solder side of component , ok C: height of tin “ 1/3 of solder side of component, reject	
08	SMT	<p style="text-align: center;">Normal</p>  <p style="text-align: center;">Jointing side</p>		
		Shortcircuit , Open circuit	Forbid	
09	Light	Quality of CSTN Display	1 , Rolling strake with visual inspection, forbid 2 , Differentness of color in viewing area with visual inspection ( full white , red , green , blue), forbid 3 , Display change with visual inspection , forbid	

10	Color Of CIE Coordinate		x	y	Drive LCD under normal condition, 25°C, 8=0 0=0 Test white, red, green blue with DMS Record
		white	±0.05	±0.05	
		Red	±0.05	±0.05	
		Green	±0.05	±0.05	
		Blue	±0.05	±0.05	
		According to the specification or sample customer have approved			
11	Brightness	In accord with product specification	Drive condition is according to specification Measure location is in Follow Picture 3, Adjust brightness instrument to zero, burrow against the surface of LCD, press "measure", record when the display is steady. (YOKOGAWA-3298)		
			 <p style="text-align: center;">Measure location</p>		
12	CR (Max)	According to specification	According to product specification Measure instrument ( DMS-501 )		
13	Response time	According to specification	According to product specification Measure instrument ( DMS-501 )		
14	Viewing angle	According to specification	According to product specification Measure instrument ( DMS-501 )		
15	Vibration, Ring	Compare with the sample customer supply	Compare with the sample customer supply when assemble		
16	Frequency Of FPC Bend	According to the use of product ( main FPC of foldaway cell phone " 6 thousand )	Measure instrument Bend angle : 150° Fix FPC in the casement when customer supply		

## 9 Handling Precautions

### 9.1 Mounting method

The LCD panel of Daxian LCD module consists of two thin glass plates with polarizers which easily be damaged. And since the module is so constructed as to be fixed by utilizing fitting holes in the printed circuit board.

Extreme care should be needed when handling the LCD modules.

### 9.2 Caution of LCD handling and cleaning

When cleaning the display surface, Use soft cloth with solvent [recommended below] and wipe lightly

- Isopropyl alcohol
- Ethyl alcohol

Do not wipe the display surface with dry or hard materials that will damage the polarizer surface.

Do not use the following solvent:

- Water
- Aromatics

Do not wipe ITO pad area with the dry or hard materials that will damage the ITO patterns

Do not use the following solvent on the pad or prevent it from being contaminated:

- Soldering flux
- Chlorine (Cl) , Sulfur (S)

If goods were sent without being silicic acid coated on the pad, ITO patterns could be damaged due to the corrosion as time goes on.

If ITO corrosion happens by miss-handling or using some materials such as Chlorine (Cl), Sulfur (S) from customer, Responsibility is on customer.

### 9.3 Caution against static charge

The LCD module uses C-MOS LSI drivers, so we recommend that you:

Connect any unused input terminal to V<sub>dd</sub> or V<sub>ss</sub>, do not input any signals before power is turned on, and ground your body, work/assembly areas, assembly equipment to protect against static electricity.

### 9.4 packing

- Module employs LCD elements and must be treated as such.
- Avoid intense shock and falls from a height.
- To prevent modules from degradation, do not operate or store them exposed direct to sunshine or high temperature/humidity

### 9.5 Caution for operation

- It is an indispensable condition to drive LCD's within the specified voltage limit since the higher voltage than the limit causes the shorter LCD life.
- An electrochemical reaction due to direct current causes LCD's undesirable deterioration, so that the use of direct current drive should be avoided.
- Response time will be extremely delayed at lower temperature than the operating temperature range and on the other hand at higher temperature LCD's show dark color in them. However those phenomena do not mean malfunction or out of order with LCD's, which will come back in the specified operation temperature.
- If the display area is pushed hard during operation, some font will be abnormally

- displayed but it resumes normal condition after turning off once.
- A slight dew depositing on terminals is a cause for electro-chemical reaction resulting in terminal open circuit.  
Usage under the maximum operating temperature, 50%Rh or less is required.

## 9.6 storage

In the case of storing for a long period of time for instance, for years for the purpose or replacement use, the following ways are recommended.

- Storage in a polyethylene bag with the opening sealed so as not to enter fresh air outside in it . And with no desiccant.
- Placing in a dark place where neither exposure to direct sunlight nor light's keeping the storage temperature range.
- Storing with no touch on polarizer surface by the anything else.  
[It is recommended to store them as they have been contained in the inner container at the time of delivery from us

## 9.7 Safety

- It is recommendable to crash damaged or unnecessary LCD's into pieces and wash off liquid crystal by either of solvents such as acetone and ethanol, which should be burned up later.
- When any liquid leaked out of a damaged glass cell comes in contact with your hands, please wash it off well with soap and water

# 10 Precaution for use

## 10.1

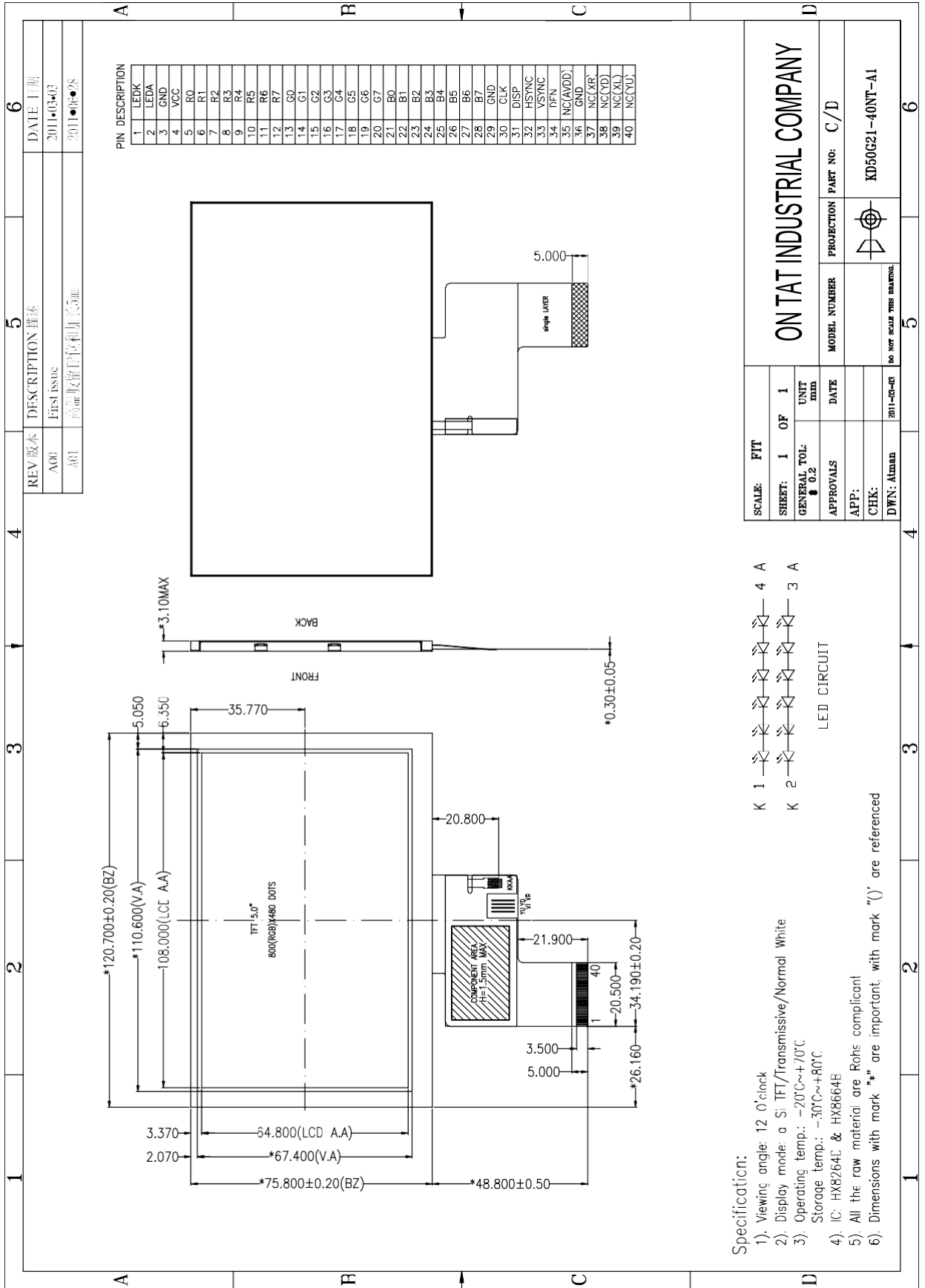
A limit sample should be provided by the both parties on an occasion when the both parties agreed its necessity. Judgment by a limit sample shall take effect after the limit sample has been established and confirmed by the both parties.

## 10.2

On the following occasions, the handing of problem should be decided through discussion and agreement between responsible of the both parties.

- When a question is arisen in this specification
  - When a new problem is arisen which is not specified in this specifications
  - When an inspection specifications change or operating condition change in customer is reported to Daxian , and some problem is arisen in this specification due to the change
  - When a new problem is arisen at the customer's operating set for sample evaluation in the customer site.
-

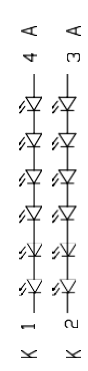
11 Dimensional Outline



Specification:

- 1). Viewing angle: 12 0'clock
- 2). Display mode: a Si TFT/Transmissive/Normal White
- 3). Operating temp.: -20°C~+70°C  
Storage temp.: -30°C~+80°C
- 4). IC: HX8264C & HX8664B
- 5). All the raw material are RoHS compliant
- 6). Dimensions with mark "\*" are important, with mark "(" are referenced

SCALE:	FIT
SHEET:	1 OF 1
GENERAL TOL:	UNIT: mm
APPROVALS:	DATE:
APP:	
CHK:	
DWN:	Alman 2011-05-05



ON TAT INDUSTRIAL COMPANY

MODEL NUMBER	PROJECTION	PART NO:
		KD50G21-40NT-A1
DO NOT SCALE THIS DRAWING		

**MANUAL DE FUNCIONAMIENTO DEL SISTEMA DE CCTV MEDIANTE  
VISION POR COMPUTADOR PARA EL CENTRO DE EDUCACION  
INICIAL N1.**

**OBJETIVOS DEL MANUAL: *Mostrar el proceso básico de instalación del sistema CCTV MEDIANTE VISIÓN POR COMPUTADOR.***

***Objetivos Específicos:***

***Analizar las librerías utilizadas para el correcto funcionamiento***

***Conocer a breves rasgos los aspectos básicos de y los implementos del sistema implementado sin mostrar la esencia energética que este lleva por protección futura.***

***Mostrar como se integran los respectivos softwares desarrollados***

***Ser un referente para mejoras y modificaciones mas compactas que brinden una mejor seguridad.***

## PARTES QUE LO COMPONEN

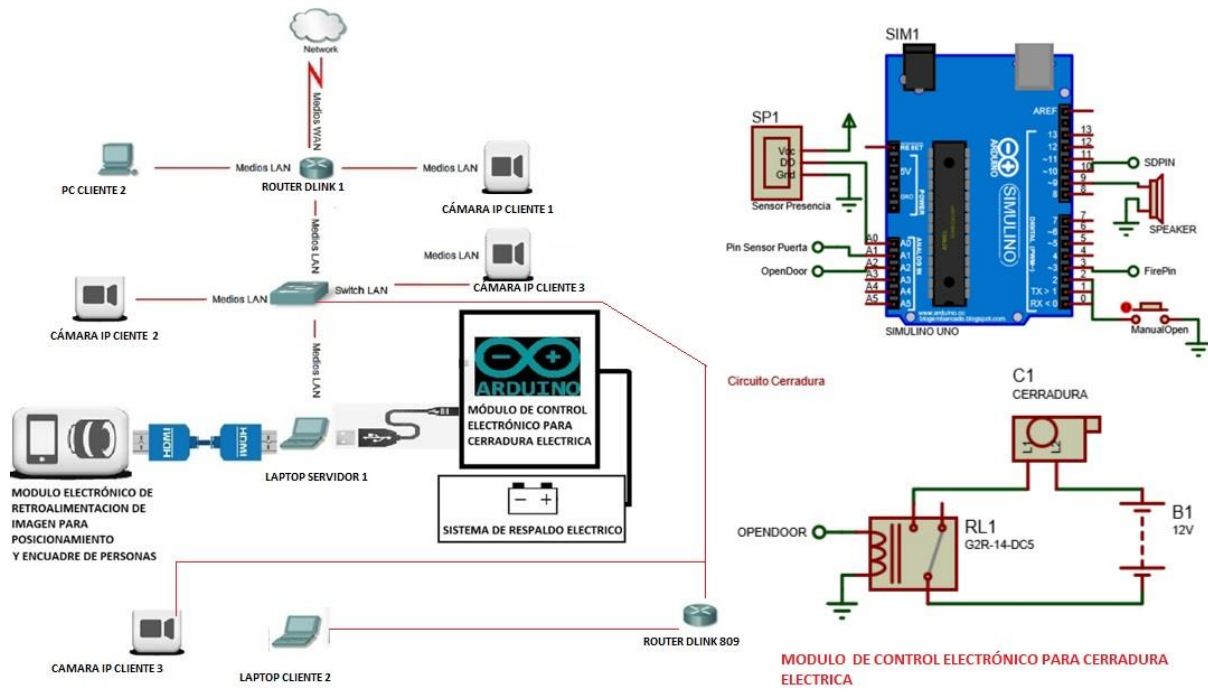


DIAGRAMA LAN DEL CIRCUITO CCTV MEDIANTE VISIÓN POR COMPUTADORA

**FIG 1 : diagrama general del sistema cctv mediante visión por computadora elaborado por Leonardo Arellano**

### 1. MODULO DE CONTROL ELECTRÓNICO PARA CERRADURA ELÉCTRICA

- SWITCH
- ACTUADOR

### 2. MODULO ELECTRÓNICO DE RETROALIMENTACIÓN DE IMAGEN PARA POSICIONAMIENTO Y ENCUADRE DE PERSONAS

- PANTALLA TFT
- CAMARA USB LOGIC
- PARLANTES

### **3. SISTEMA DE RESPALDO ELÉCTRICO**

- **BATERIA**
- **CONTROLADOR DE CARGA**
- **REGLETA DE ENERGIZACIÓN**

### **4. LAPTOP SERVIDOR 1**

- **CARPETA EJECUTABLE DE SERVIDOR**

### **5. LAPTOP CLIENTE 1**

- **CARPETA EJECUTABLE CLIENTE 1**

### **6. PC CLIENTE 2**

- **CARPETA EJECUTABLE CLIENTE 2**

### **7. COMUNICACIONES**

- **ROUTER DLINK 1 MODELO**
- **ROUTER DLINK 2 DIR 809**

### **8. SENSORES DE VISIÓN**

- **CAMARA IP CLIENTE 1**
- **CAMARA IP CLIENTE 2**
- **CAMARA IP CLIENTE 3**



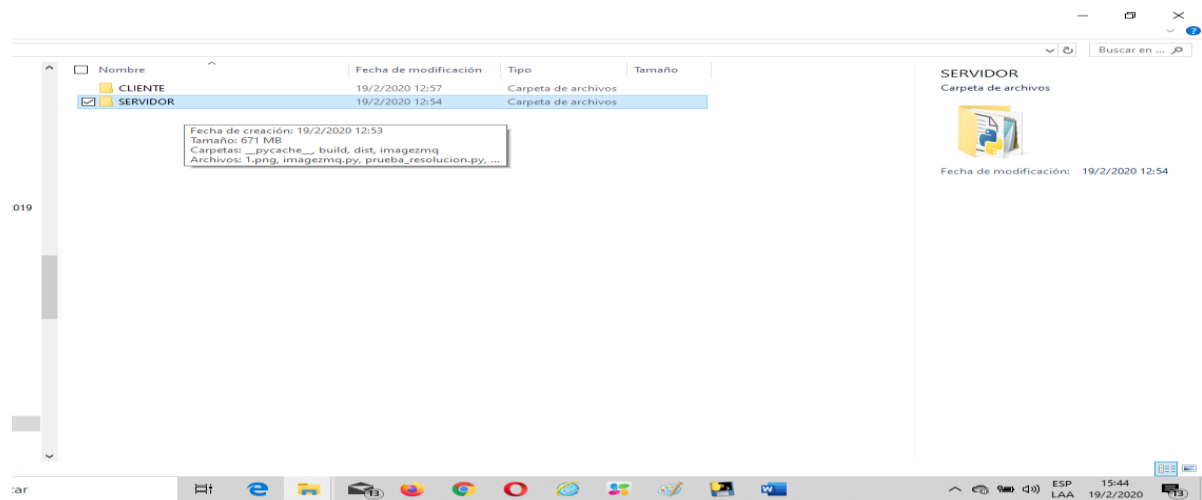
## **PASOS PARA FUNCIONAMIENTO.**

### **PRIMERO:**

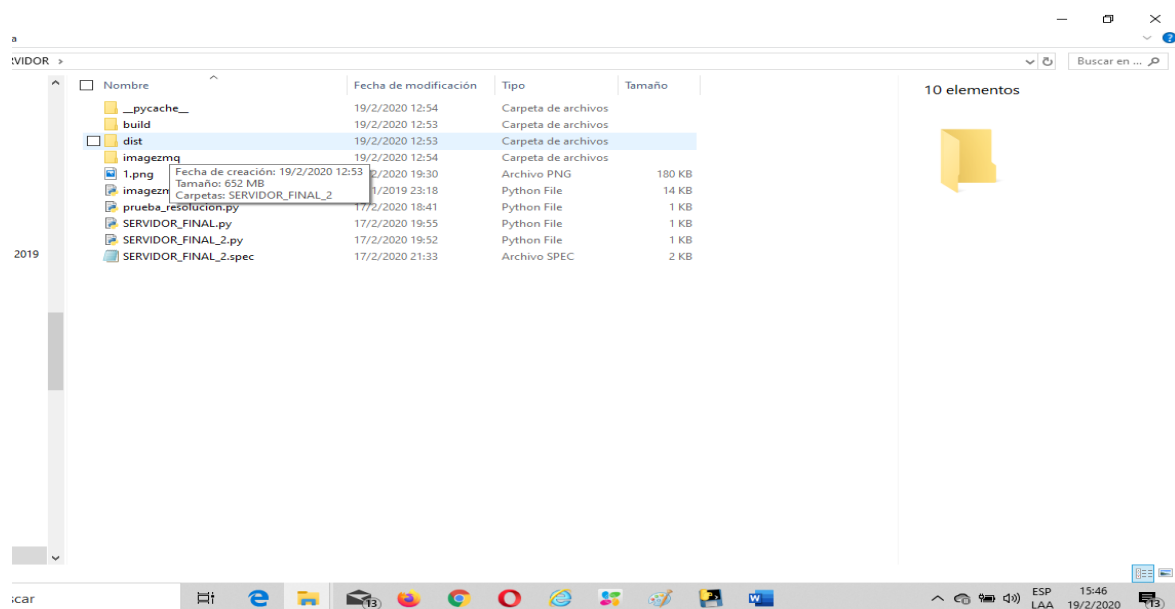
**CONECTE LOS DISPOSITIVOS Y MODULOS COMO MUESTRA Fig.1**

### **SEGUNDO:**

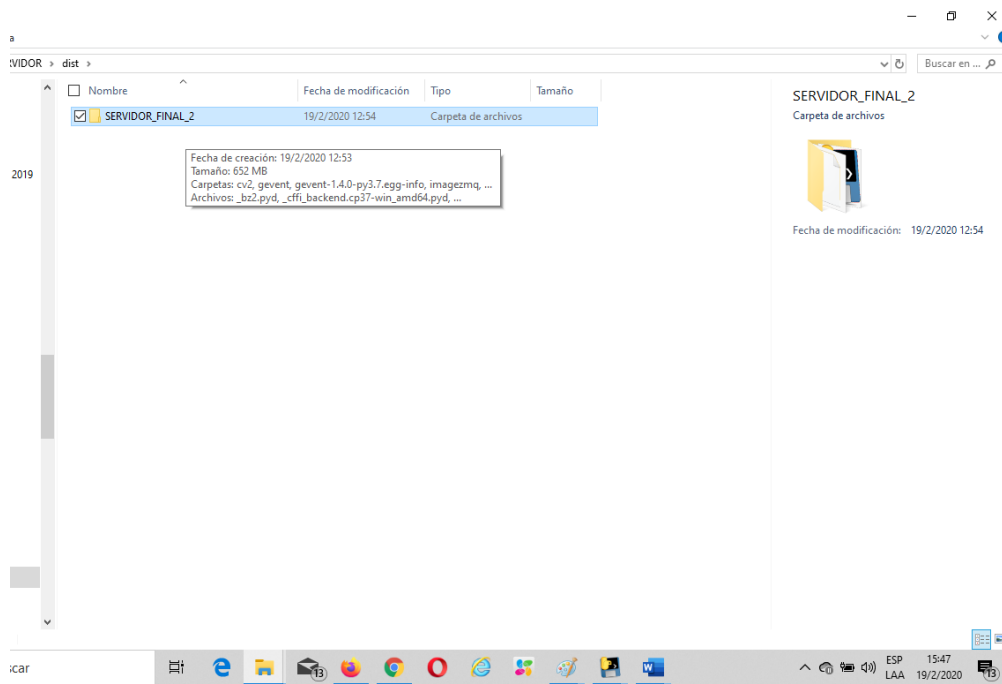
**COPIE LA CARPETA SERVIDOR a la computadora denominada LAPTOP SERVIDOR 1**



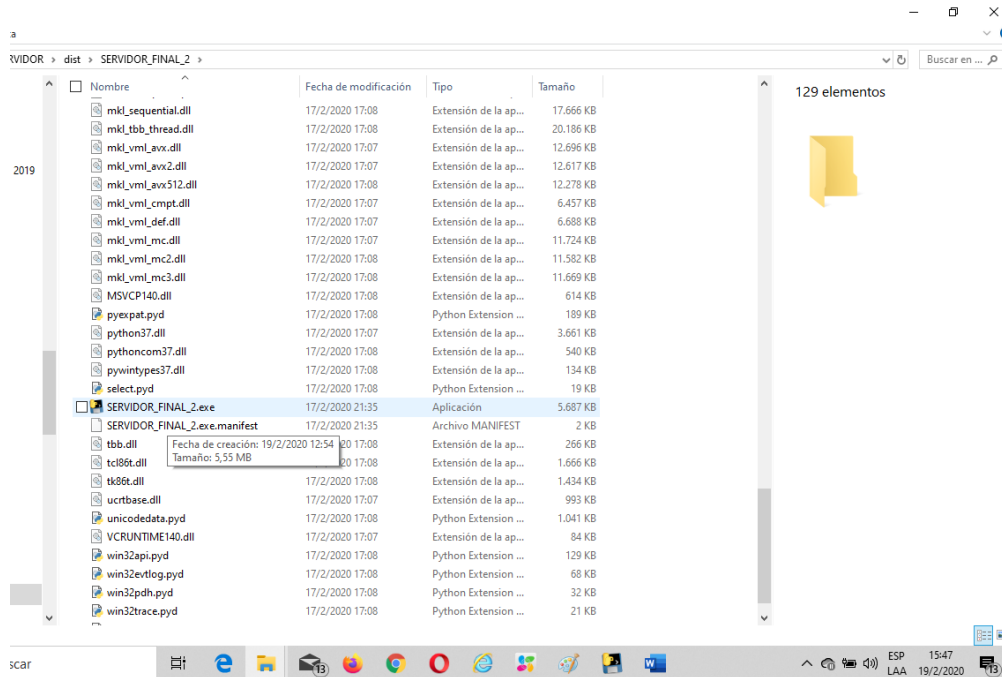
## **INGRESE A LA CARPETA dist**



## Ingrese a la carpeta **SERVIDOR FINAL**

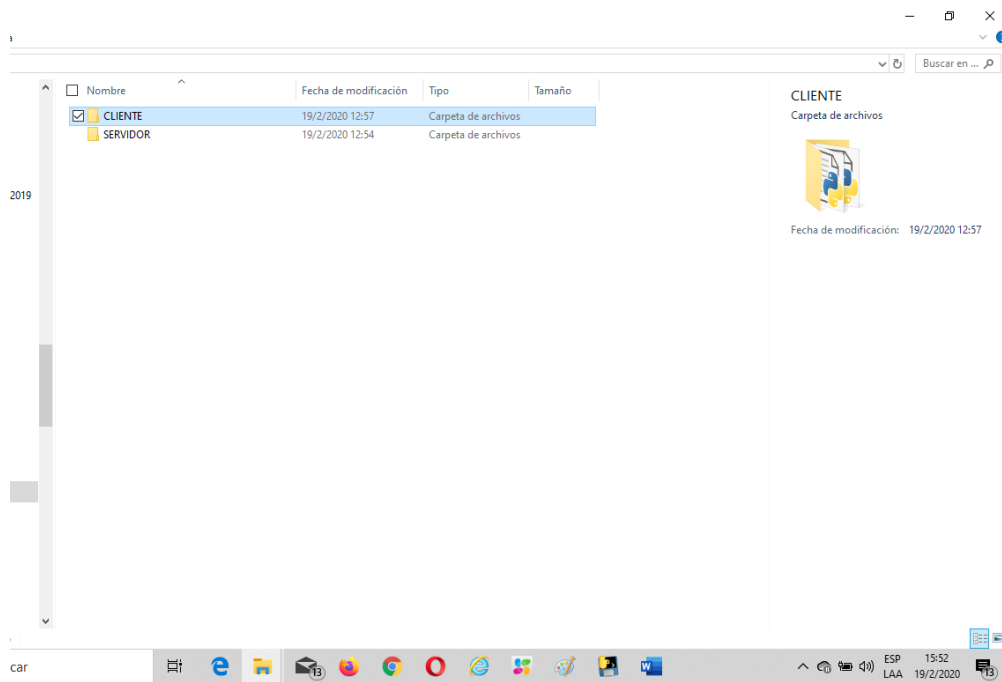


## DE DOBLE CLICK EN EL ARCHIVO **SERVIDOR FINAL 2**

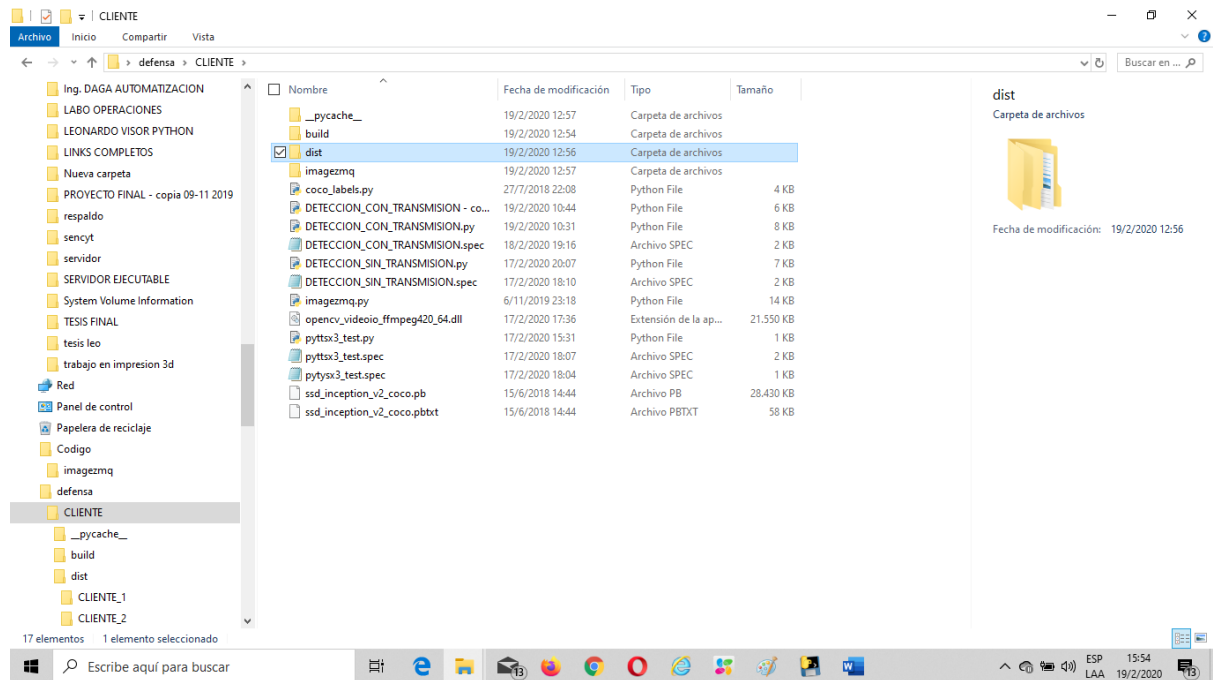


## TERCERO:

### COPIE DE LA CARPETA CLIENTE

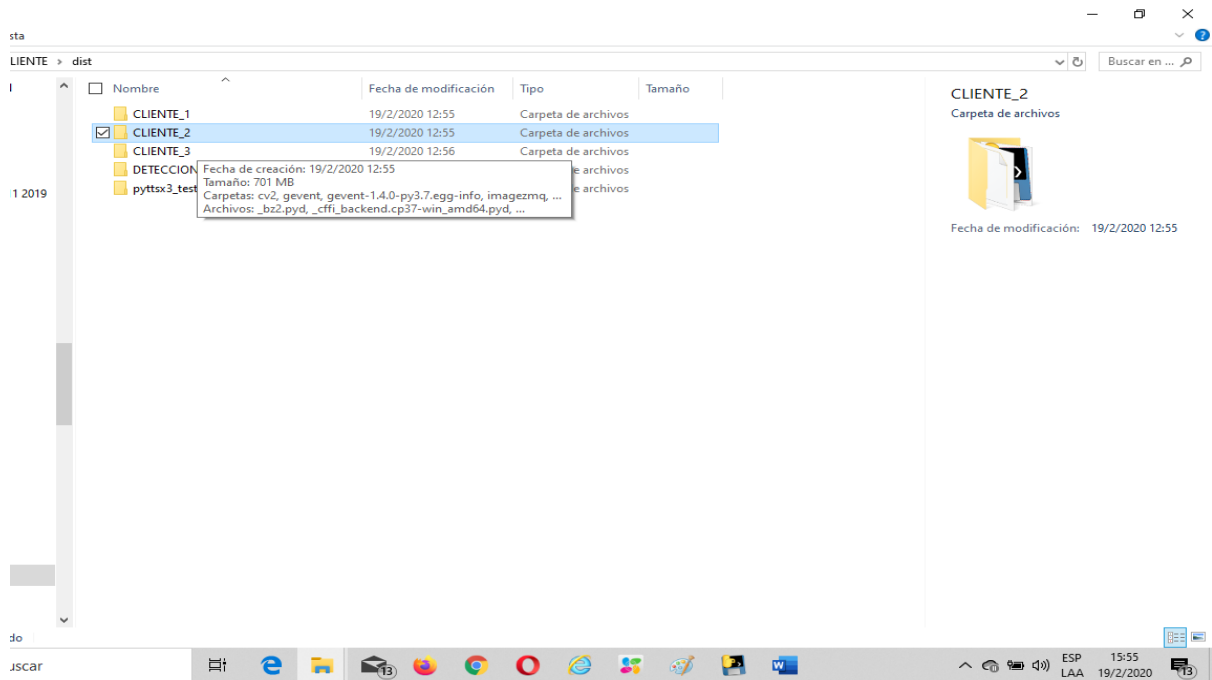


### la carpeta dist

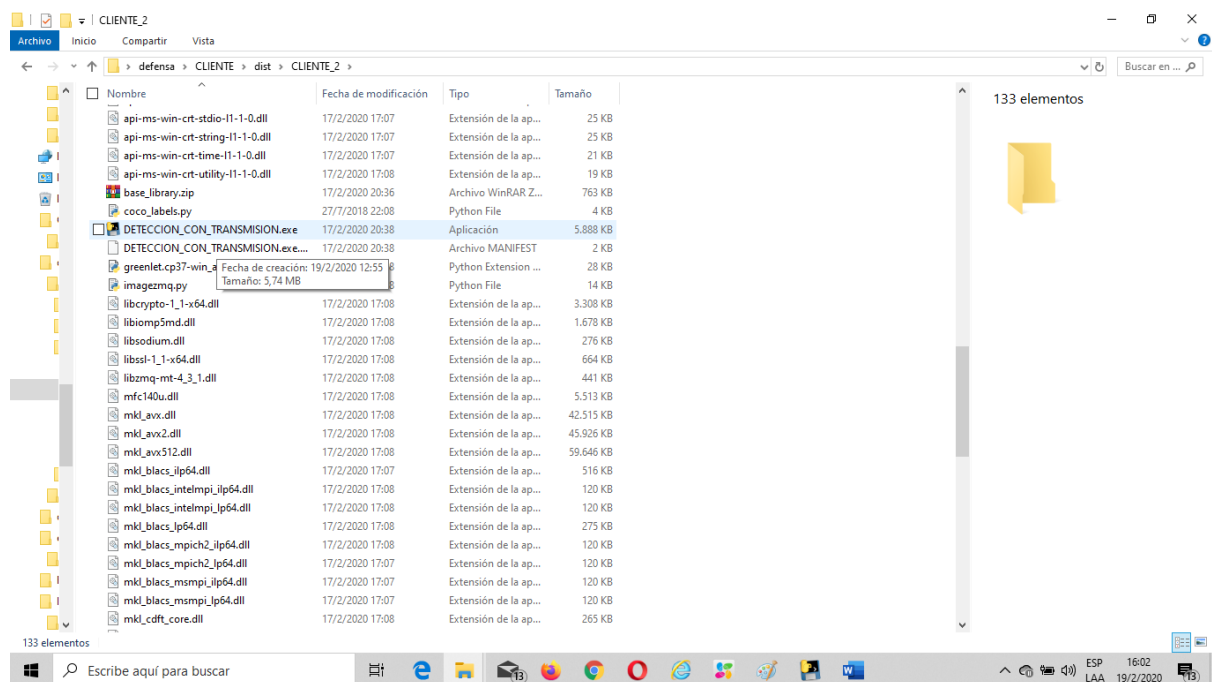


## CUARTO:

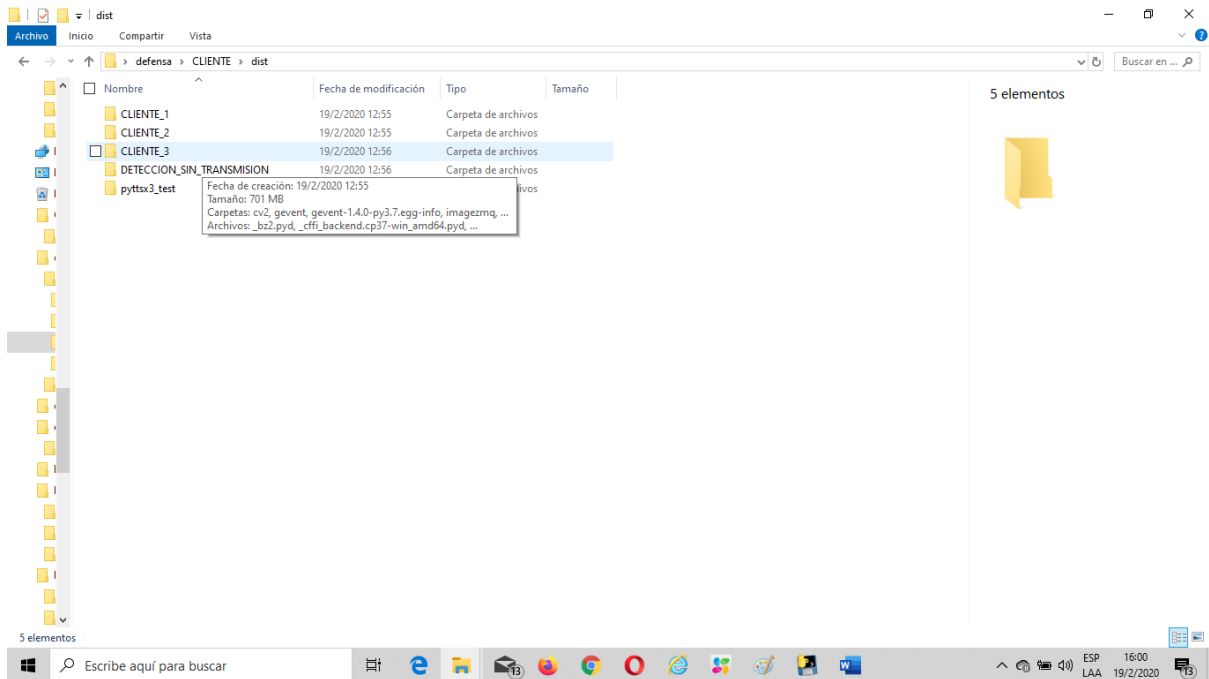
**Copie La carpeta CLIENTE 2 a la PC CLIENTE 2 del diagrama de la Fig 1. Y luego ábralo.**



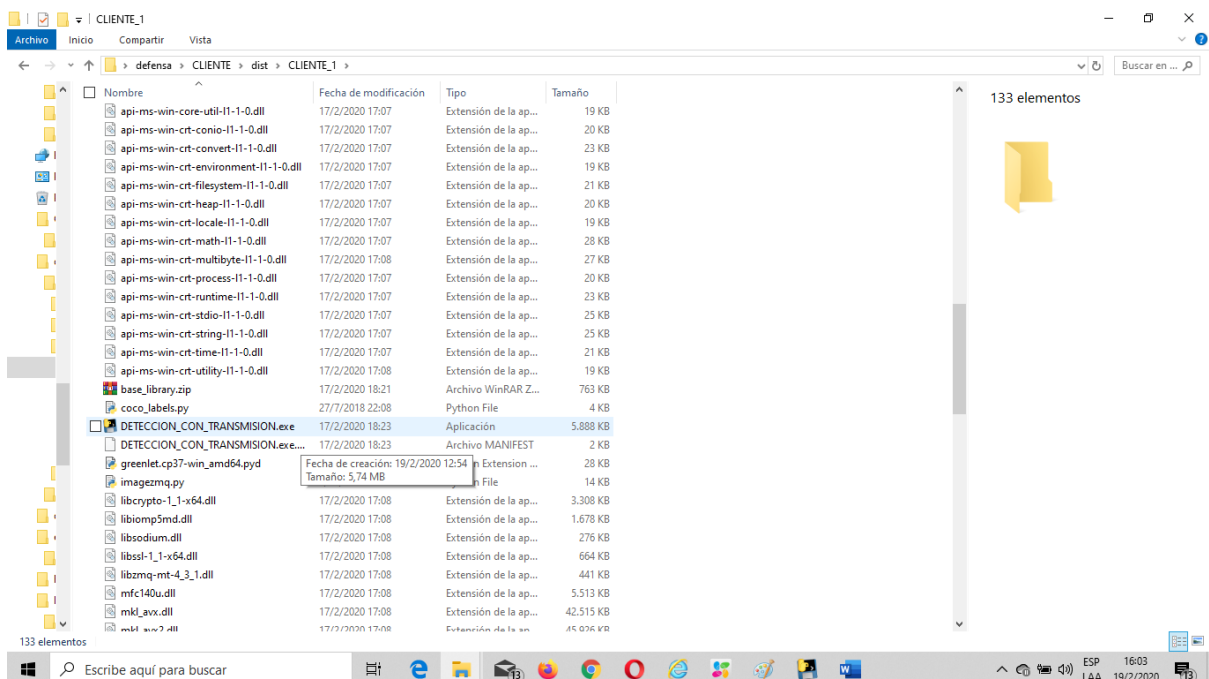
**Ejecute dando doble clic el archivo detección con transmisión**



**Copie la CARPETA CLIENTE 1 Y CLIENTE 3 A LA LAPTOP CLIENTE 2 del diagrama de la fig. 1. Y ábralos.**



**Ejecute los siguientes archivos de cada carpeta en la laptop CLIENTE 2.**

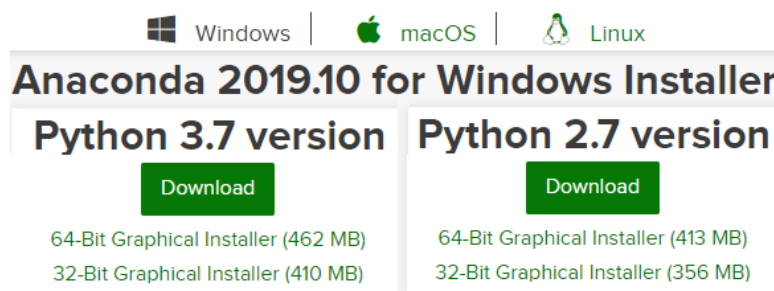


## **QUINTO: EN LA LAPTOP SERVIDOR 1 REALIZAR LOS SIGUIENTES SUBPROCESO**

- **Descargar e instalar Anaconda**
- **Creación del entorno virtual Face\_Recognition**
- **Instalar el IDE Spyder**
- **Instalación de paquetes requeridos**
  - **Descargar Anaconda**

**Anaconda es una distribución de Python, que instala Python y un conjunto de paquetes útiles para el análisis científico. Brinda una gran cantidad de funcionalidades que permiten desarrollar aplicaciones en forma más eficiente, rápida y sencilla. Es libre y de código abierto, con gran cantidad de documentación y una gran comunidad.**

**Se descarga del Home de Anaconda en la sección Dowload. Se debe elegir el tipo de sistema operativo, la versión más reciente y el tipo de procesador de la PC en la que será instalado.**



**Figura 2. Instalación de Python y Anaconda**

**Fuente: [www.anaconda.com/distribution](http://www.anaconda.com/distribution)**

**Ejecutar el archivo descargado y seguir las instrucciones del asistente de instalación. Se requiere 1GB de disco duro aproximadamente para la instalación.**

- **Creación del entorno virtual `face_recognition`**

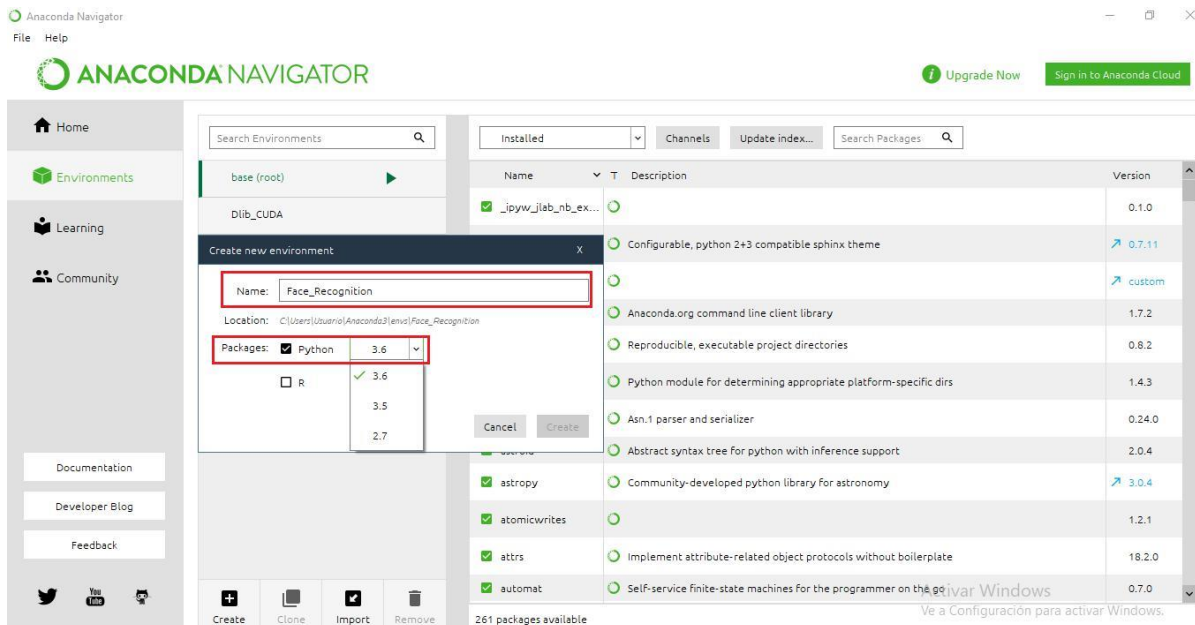
**Un entorno virtual es un espacio independiente a la instalación local, con el objetivo de aislar los recursos y librerías. Gracias a este concepto se puede tener distintos entornos virtuales con diferentes versiones de Python o de una librería concreta. Las razones por las que deberías usar entornos virtuales son:**

**Organización:** es conveniente tener controladas las librerías y las versiones con las que trabaja para que no existan problemas de incompatibilidades, y para ello, es mejor organizar todo en entornos virtuales, así, no tener la instalación de Python con multitud de librerías instaladas.

**Estabilidad:** es un hecho que las librerías cambian de versión muy a menudo, esto puede ocasionar que los programas ya no funcionen.

**Trabajar con diferentes versiones de Python:** existen muchas aplicaciones en la versión 2.7 de Python y algunas no cuentan con soporte para Python 3. Creando distintos entornos virtuales se puede cambiar de versión de Python en dos líneas de código.

**Inicialmente se ejecuta la interfaz gráfica de usuario (GUI) Anaconda Navigator. En la sección Environments seleccionar Create:**



**Figura. 3 creación del entorno virtual face recognition**

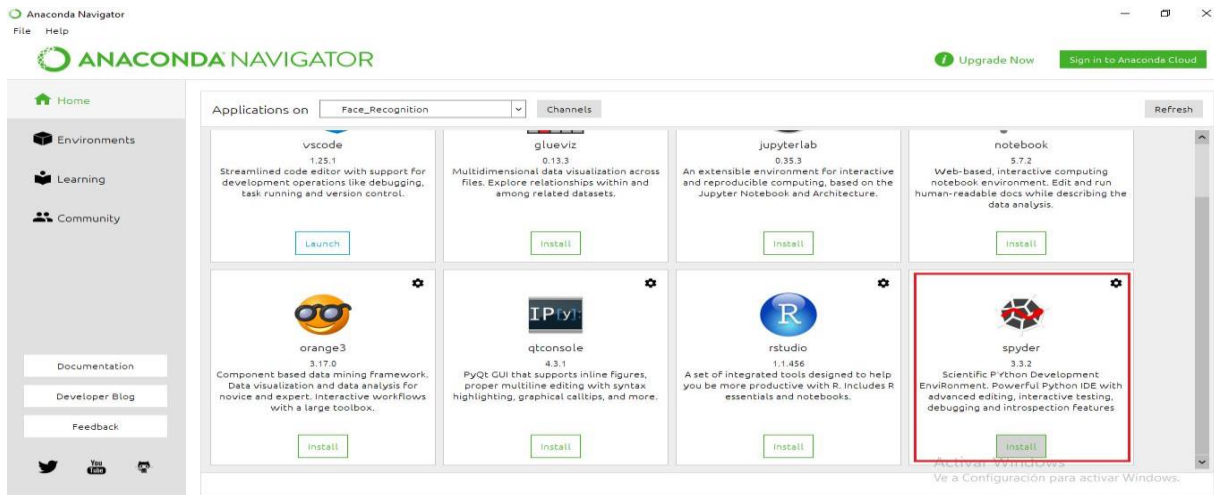
**Fuente: Autor**

**Definir el nombre del nuevo entorno virtual y la versión de Python a utilizar. En la ventana del lado derecho Descriptions, se muestran las librerías instaladas en el entorno virtual con sus versiones.**

○ **Instalar el IDE Spyder**

**En la sección Home de Anaconda instalar el IDE Spyder dentro del entorno virtual creado. Spyder es un potente entorno de desarrollo interactivo para el lenguaje Python. Posee funciones avanzadas de edición, pruebas interactivas, depuración e introspección y un entorno informático numérico. Gracias al soporte de IPython (intérprete interactivo mejorado de Python) y bibliotecas populares de Python como NumPy, SciPy o matplotlib (trazado interactivo 2D / 3D).**





**Figura 4. Instalación del IDE Spyder**

**Fuente: Autor**

**Verificar en terminal de Python los paquetes instalados en entorno virtual. Retorne a la ventana Environments. Acceda a la ventana de comandos de Windows dando click sobre la flecha en Face\_Recognition, seleccione Open Terminal y ejecute el comando: conda list**

```

C:\Windows\system32\cmd.exe
(Face_Recognition) C:\Users\Usuario>conda list
# packages in environment at C:\Users\Usuario\Anaconda3\envs\Face_Recognition:
#
# Name            Version           Build    Channel
certifi           2018.10.15        py36_0
pip               18.1              py36_0
python            3.6.7             h33f27b4_1
setuptools        40.5.0            py36_0
vc                14.1              h0510ff6_4
vs2015_runtime   14.15.26706      h3a45250_0
wheel             0.32.2            py36_0
wincertstore      0.2               py36h7fe50ca_0
  
```

**Figura 5. Verificación de paquetes instalados en el entorno virtual**

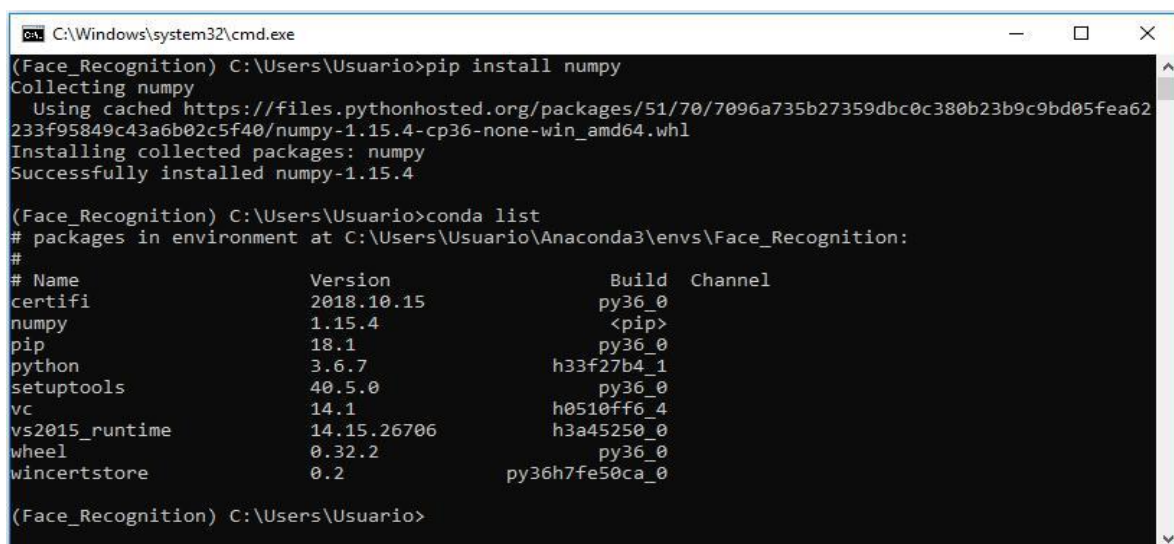
**Fuente: Autor**

- **Instalación de paquetes requeridos**

**La instalación de paquetes de Python se realiza a través de la ventana de comandos de Windows en el entorno virtual *Face\_recognition* creado en el paso anterior. Para el desarrollo de la aplicación de reconocimiento facial bajo Python, OpenCV y técnicas de Deep Learning, se requiere la instalación de las siguientes librerías externas:**

***NUMPY* es una extensión de Python, que le agrega mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices. Versión 1.15.4.**

***pip install numpy***



```
C:\Windows\system32\cmd.exe
(Face_Recognition) C:\Users\Usuario>pip install numpy
Collecting numpy
  Using cached https://files.pythonhosted.org/packages/51/70/7096a735b27359dbc0c380b23b9c9bd05fea62233f95849c43a6b02c5f40/numpy-1.15.4-cp36-none-win_amd64.whl
Installing collected packages: numpy
Successfully installed numpy-1.15.4

(Face_Recognition) C:\Users\Usuario>conda list
# packages in environment at C:\Users\Usuario\Anaconda3\envs\Face_Recognition:
#
# Name                    Version            Build             Channel
certifi                   2018.10.15         py36_0
numpy                     1.15.4             <pip>
pip                       18.1               py36_0
python                   3.6.7              h33f27b4_1
setuptools                40.5.0             py36_0
vc                        14.1               h0510ff6_4
vs2015_runtime            14.15.26706       h3a45250_0
wheel                     0.32.2             py36_0
wincertstore              0.2                py36h7fe50ca_0

(Face_Recognition) C:\Users\Usuario>
```

**Figura. 3.1. Instalación de numpy en el entorno virtual**

**Fuente: Autor**

***OPENCV (Open Source Computer Vision)* es una librería software open-source de visión artificial y machine learning. OpenCV provee una infraestructura para aplicaciones de visión artificial. Version 3.2.0.**

***conda install -c conda-forge opencv=3.2.0***

```

C:\Windows\system32\cmd.exe - conda install -c conda-forge opencv=3.2.0
(Face_Recognition) C:\Users\Usuario>conda install -c conda-forge opencv=3.2.0
Solving environment: done

## Package Plan ##

  environment location: C:\Users\Usuario\Anaconda3\envs\Face_Recognition

  added / updated specs:
  - opencv=3.2.0

The following packages will be downloaded:

package | build | size | channel
-----|-----|-----|-----
icu-58.2 | vc14_0 | 21.8 MB | conda-forge
mkl_fft-1.0.6 | py36_0 | 121 KB | conda-forge
mkl_random-1.0.2 | py36_0 | 267 KB | conda-forge
libpng-1.6.34 | vc14_0 | 547 KB | conda-forge
qt-5.6.2 | vc14_1 | 55.6 MB | conda-forge
opencv-3.2.0 | np113py36_204 | 92.0 MB | conda-forge
zlib-1.2.11 | vc14_0 | 119 KB | conda-forge

```

**Figura. 6. Instalación de Open CV**

**Fuente: Autor**

***DLIB es un moderno kit de herramientas de C++ que contiene algoritmos de aprendizaje de máquina (machine learning) y herramientas para crear software complejo en C ++. Es una librería “para resolver problemas del mundo real”. Se utiliza en la industria y en el mundo académico en una amplia gama de dominios, incluyendo la robótica, dispositivos integrados, teléfonos móviles y grandes entornos de computación de alto rendimiento. Las licencias de código abierto de Dlib permiten usarlo en cualquier aplicación, de forma gratuita. Versión 19.9.0***

***conda install -c conda-forge dlib***

```

C:\Windows\system32\cmd.exe - conda install -c conda-forge dlib
(Face_Recognition) C:\Users\Usuario>conda install -c conda-forge dlib
Solving environment: done

## Package Plan ##

  environment location: C:\Users\Usuario\Anaconda3\envs\Face_Recognition

  added / updated specs:
  - dlib

The following packages will be downloaded:

package | build | size | channel
-----|-----|-----|-----
dlib-19.9 | np111py36_0 | 2.4 MB | conda-forge
opencv-3.2.0 | np111py36_204 | 92.0 MB | conda-forge
opensl-1.0.2p | hfa6e2cd_1001 | 5.4 MB | conda-forge
numpy-1.11.3 | py36h53ece5f_11 | 11 KB | conda-forge
numpy-base-1.11.3 | py36h8128ebf_11 | 3.3 MB | conda-forge
-----|-----|-----|-----
Total: | | 103.0 MB |

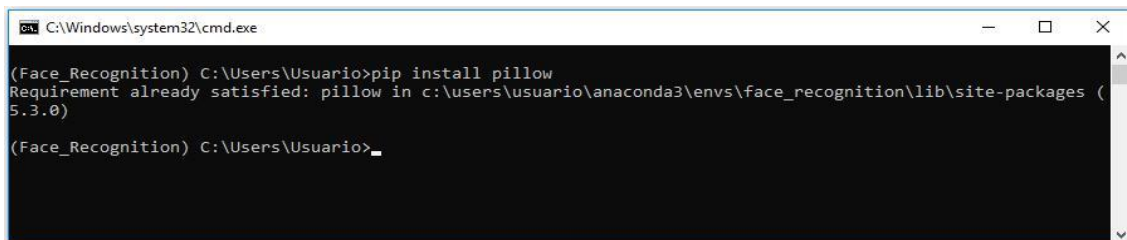
```

**Figura. 7. Instalación de la librería pillow**

**Fuente: Autor**

***PILLOW es una librería gratuita que permite la edición de imágenes directamente desde Python. Soporta una variedad de formatos, incluidos los más utilizados como GIF, JPEG y PNG. Es una bifurcación de PIL Python Imaging Library (PIL) para versiones 3.x de Python. Versión 5.3.0.***

***pip install pillow***



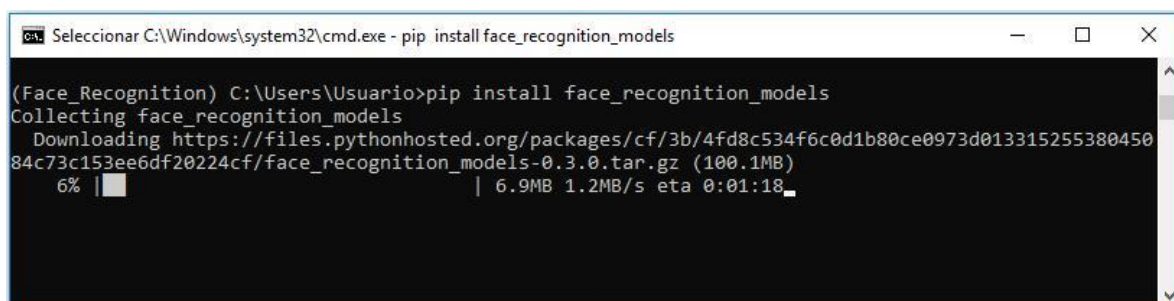
```
C:\Windows\system32\cmd.exe
(Face_Recognition) C:\Users\Usuario>pip install pillow
Requirement already satisfied: pillow in c:\users\usuario\anaconda3\envs\face_recognition\lib\site-packages (5.3.0)
(Face_Recognition) C:\Users\Usuario>_
```

***Figura. 8. Instalación de la librería pillow***

***Fuente: Autor***

***Face\_recognition. Biblioteca para el reconocimiento facial basada en técnicas de Deep learning que permite definir el vector de características para cada rostro. Basado en dlib, un moderno kit de herramientas de C++ que contiene varios algoritmos de aprendizaje automático que ayudan a escribir aplicaciones sofisticadas basadas en C ++. Versión 1.2.3***

***pip install face\_recognition\_models***



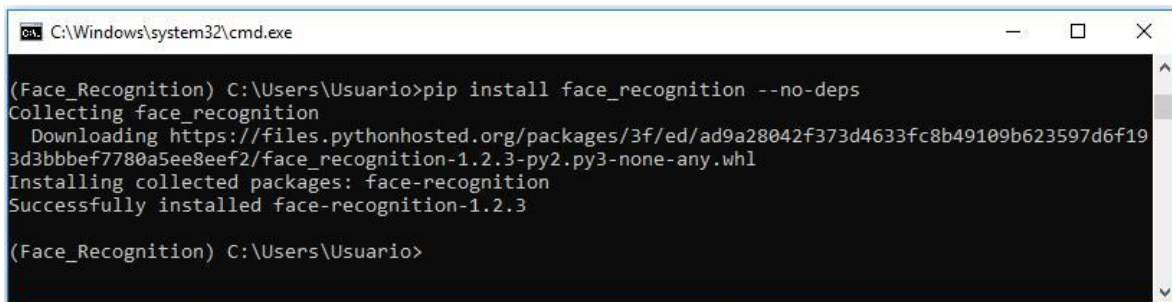
```
Seleccionar C:\Windows\system32\cmd.exe - pip install face_recognition_models
(Face_Recognition) C:\Users\Usuario>pip install face_recognition_models
Collecting face_recognition_models
  Downloading https://files.pythonhosted.org/packages/cf/3b/4fd8c534f6c0d1b80ce0973d01331525538045084c73c153ee6df20224cf/face_recognition_models-0.3.0.tar.gz (100.1MB)
  6% |█| 6.9MB 1.2MB/s eta 0:01:18_
(Face_Recognition) C:\Users\Usuario>_
```

***Figura. 9. Instalación de la biblioteca face\_recognition***

***Fuente: Autor***

**Agregar dependencias para la biblioteca `face_recognition`.**

**`pip install face_recognition --no-deps`**



```
C:\Windows\system32\cmd.exe

(Face_Recognition) C:\Users\Usuario>pip install face_recognition --no-deps
Collecting face_recognition
  Downloading https://files.pythonhosted.org/packages/3f/ed/ad9a28042f373d4633fc8b49109b623597d6f193d3bbbef7780a5ee8eef2/face_recognition-1.2.3-py2.py3-none-any.whl
Installing collected packages: face-recognition
Successfully installed face-recognition-1.2.3

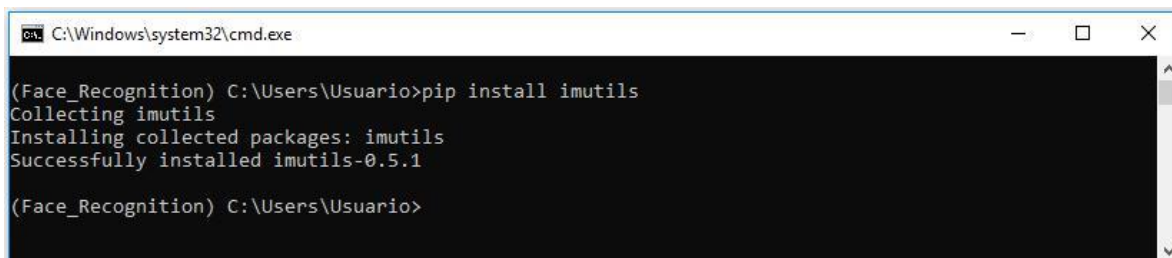
(Face_Recognition) C:\Users\Usuario>
```

**Figura. 10. Instalación de dependencias de `face_recognition`**

**Fuente: Autor**

***Imutils es un paquete de funciones elementales de procesamiento de imágenes básicas, tales como la traslación, rotación, cambio de tamaño, esqueletización, y la visualización de imágenes. Versión 0.5.1.***

**`pip install imutils`**



```
C:\Windows\system32\cmd.exe

(Face_Recognition) C:\Users\Usuario>pip install imutils
Collecting imutils
Installing collected packages: imutils
Successfully installed imutils-0.5.1

(Face_Recognition) C:\Users\Usuario>
```

**Figura. 11. Instalación de librería `imutils`**

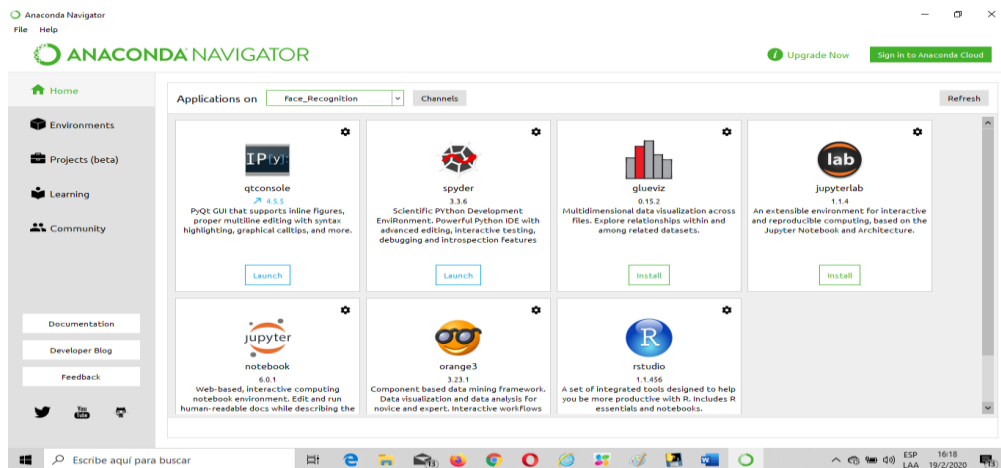
**Fuente: Autor**

***Con la instalación de las librerías anteriores, el software Python se encuentra listo para el desarrollo de aplicaciones de reconocimiento facial. Como ya se anotó el lenguaje de programación elegido es de distribución gratuita, así como todas sus librerías, es un lenguaje moderno que facilita la programación, detección de errores y dispone de una amplia documentación y comunidad en línea para el desarrollo de aplicaciones y resolución de errores. Estas son las***

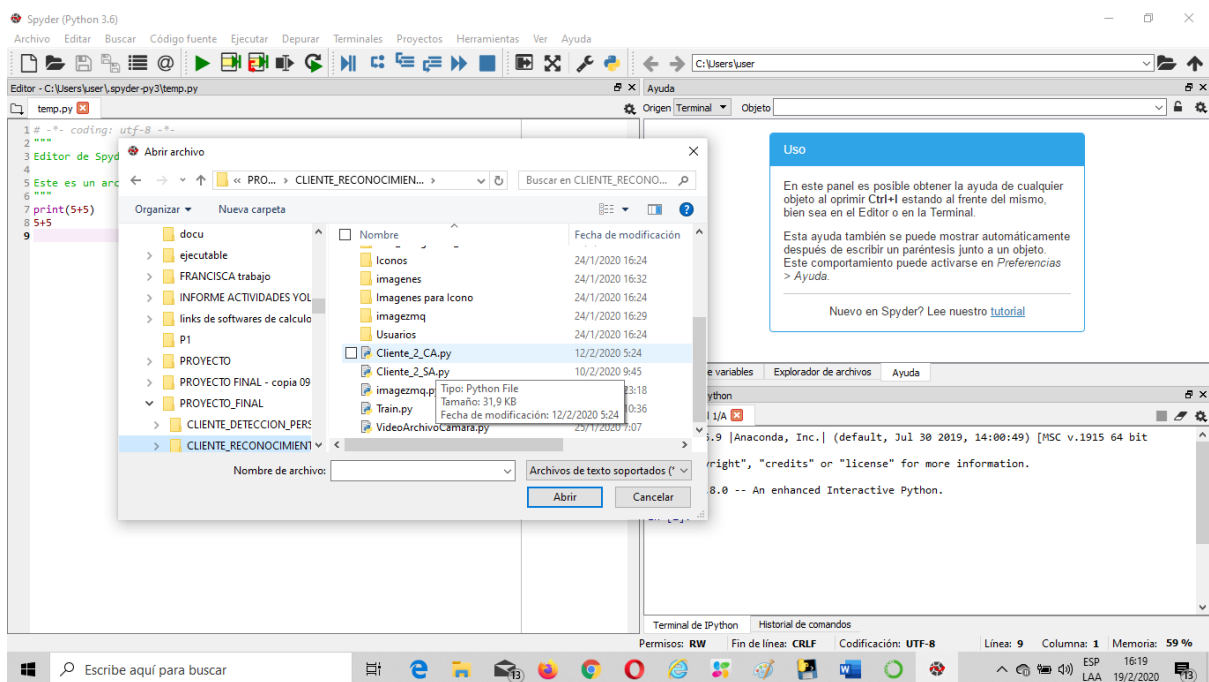
**razones por las que fue elegido como software de programación en el presente proyecto.**

**SEXTO:**

**EJECUTAR DENTRO DE FACE RECOGNITION SPYDER EN EL BOTON QUE DICE LAUNCH**

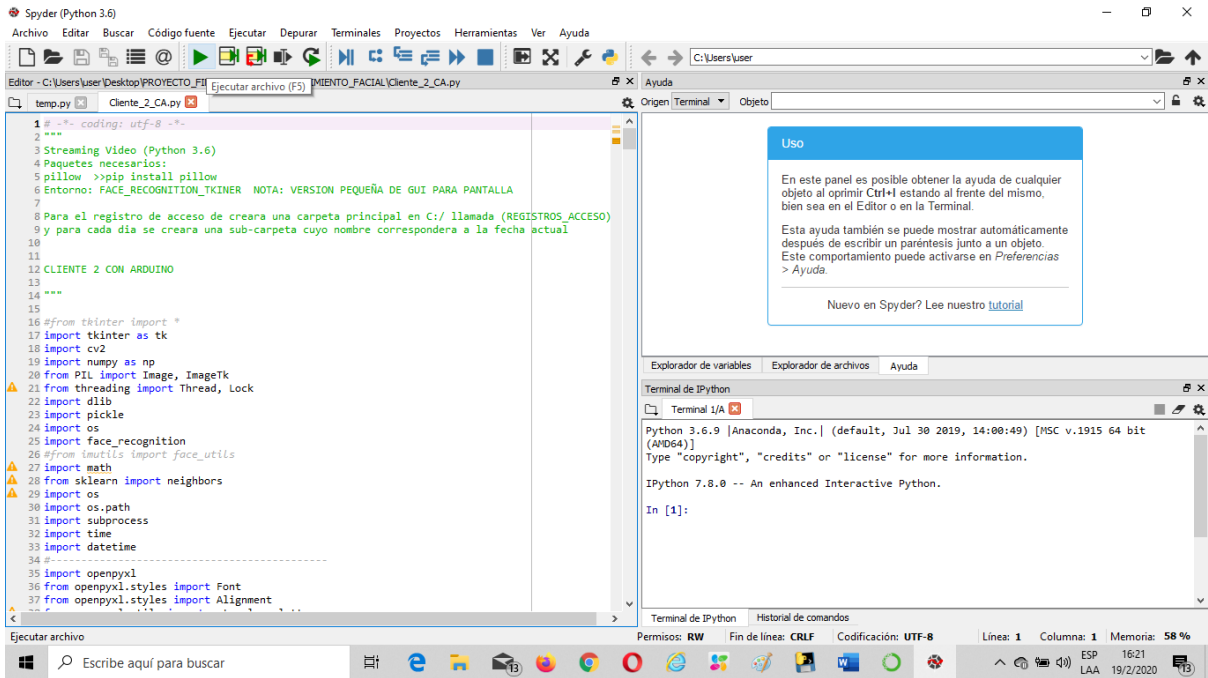


**ABRIR EL SCRIPT Cliente\_2\_CA.PY**

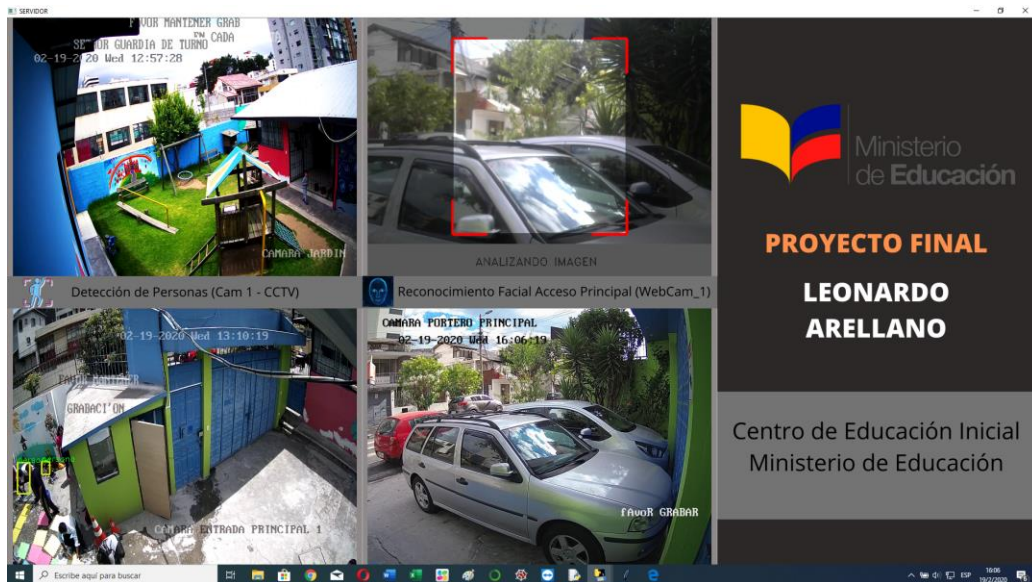


# SEPTIMO

## EJECUTAR MEDIANTE LA TECLA F5



## APARECERA LA SIGUIENTE INTERFAZ



## DECLARACIÓN Y AUTORIZACIÓN

Yo, Leonardo Rubén Arellano Arellano, con CI 1718816612 autor del trabajo de graduación:  
Tema: **SISTEMA DE VIDEO VIGILANCIA MEDIANTE VISIÓN POR COMPUTADOR PARA EL CENTRO DE EDUCACIÓN INICIAL N°1 DEL MINISTERIO DE EDUCACIÓN,**  
previo la obtención del título de **INGENIERO EN ELECTRÓNICA DIGITAL Y TELECOMUNICACIONES** en la UNIVERSIDAD TECNOLÓGICA ISRAEL.

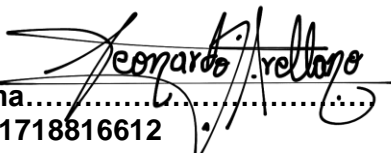
1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de difundir el respectivo trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de graduación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Quito, 04 de mayo del 2020.

**Atentamente.**

Firma.....  
C.I. 1718816612





MINISTERIO  
DE EDUCACIÓN



EL  
GOBIERNO  
DE TODOS

MANUAL DE FUNCIONAMIENTO DEL SISTEMA DE CCTV MEDIANTE  
VISION POR COMPUTADOR PARA EL CENTRO DE EDUCACIÓN INICIAL N° 1  
INICIAL N° 1  
**CERTIFICACIÓN**

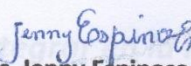
Quito, 27 de febrero de 2020

**OBJETIVOS:** Implementar un sistema de video vigilancia mediante algoritmos de machine Learning para el centro de educación inicial N.1.

A petición verbal del Señor Leonardo Rubén Arellano con número de cédula 1718816612, Certifico que el Proyecto **SISTEMA DE VIDEO VIGILANCIA MEDIANTE VISIÓN POR COMPUTADOR**, se encuentra en funcionamiento en las instalaciones del Centro de Educación Inicial No. 1, cuyos componentes electrónicos son muy indispensables para el proceso de Seguridad Institucional.

Determinar la situación actual de seguridad del establecimiento  
Es todo cuanto puedo certificar en honor a la verdad, el interesado puede hacer uso de la presente certificación en la forma que creyere más conveniente.

Atentamente,

  
**MSc. Jenny Espinosa Quintana**  
**DIRECTORA (E) DEL CENTRO DE EDUCACIÓN INICIAL Nro.1**





## Turnitin Informe de Originalidad

Procesado el: 27-feb.-2020 09:20 -05  
Identificador: 1265236959  
Número de palabras: 20651  
Entregado: 1

Índice de similitud  
**5%**

Similitud según fuente	
Internet Sources:	5%
Publicaciones:	0%
Trabajos del estudiante:	N/A

Tesis L.Arellano PorGuarquila  
Ramon Angel David

2% match (Internet desde 17-sept.-2017)

[http://www.contratacion.euskadi.eus/w32-1084/es/contenidos/anuncio\\_contratacion/expjaso3954/es\\_doc/adjuntos/pliego\\_bases\\_tecni\\_cas2.pdf](http://www.contratacion.euskadi.eus/w32-1084/es/contenidos/anuncio_contratacion/expjaso3954/es_doc/adjuntos/pliego_bases_tecni_cas2.pdf)

1% match (Internet desde 28-jul.-2012)

<http://www.sabetodo.com/documentos/algoritmos-objetos-dinamicos-290108.doc>

1% match (Internet desde 26-jul.-2019)

<https://docplayer.es/26824183-Reconocimiento-facial.html>

1% match (Internet desde 27-feb.-2019)

<https://www.devacademy.es/entornos-virtuales-en-python-anaconda>

1% match (Internet desde 04-ene.-2020)

<https://upcommons.upc.edu/bitstream/handle/2117/115538/maria-font-memoria-annex.pdf?isAllowed=y&sequence=1>

i RESUMEN Esta investigación cuyo objetivo es implementar un sistema de video vigilancia mediante visión por computador, se presenta como un proyecto factible que propone una solución a la falta de un circuito de CCTV para contrarrestar problemas de seguridad para la entidad educativa cuyo problema se abordó con una investigación aplicada en base a entrevistas a los directivos del establecimiento. Como en la mayoría de entidades públicas el ámbito económico es una limitante en el desarrollo de proyectos se optó por utilizar las nuevas tendencias tecnológicas enfocadas al área de inteligencia artificial, las cuales implican una optimización del uso de recursos, universalidad y mayor confiabilidad. Para este caso se optó por usar la visión por computador para estructurar un CCTV con aplicaciones determinadas dentro del área mencionada como reconocimiento facial y detección de personas, para lo cual existe una base de datos adquiridas mediante técnicas de machine learning, deep learning y redes neuronales, los cuales generan procesos de análisis de imágenes y creación de bases de datos dentro del CPU de un computador. El software que se elaboró para el circuito CCTV con reconocimiento facial y detección de personas es funcional para una cámara, utiliza dos pcs por el nivel de procesamiento que este genera, el cual esta acoplado a un sistema electrónico mixto, que controla una cerradura magnética. El software de reconocimiento facial y el software de detección de personas está vinculado a un servidor en el que ejecuta otro aplicativo desarrollado en Python para que se muestre al guardia de turno solo las imágenes, sin que este tenga acceso a la manipulación o modificación de las cámaras o el sistema en general. PALABRAS CLAVES: Apertura Automática, Tensorflow, Python, Reconocimiento Facial, Video Vigilancia, Visión por Computador. ii ABSTRACT This research, whose objective is to implement a video surveillance system using computer vision, is presented as a feasible project that proposes a solution to the