

	UNIVERSIDAD FRANCISCO DE PAULA SANTANDER OCAÑA			
	Documento	Código	Fecha	Revisión
	FORMATO HOJA DE RESUMEN PARA TRABAJO DE GRADO	F-AC-DBL-007	08-07-2021	B
	Dependencia	Aprobado		Pág.
DIVISIÓN DE BIBLIOTECA	SUBDIRECTOR ACADEMICO		1(379)	

RESUMEN – TRABAJO DE GRADO

AUTORES	Juan Pablo Mena Santos Wilson Javier Arias Cardona		
FACULTAD	Facultad de ingenierías		
PLAN DE ESTUDIOS	Ingeniería de sistemas		
DIRECTOR	Byron Cuesta Quintero		
TÍTULO DE LA TESIS	Diseño de un prototipo de reconocimiento facial utilizando técnicas de inteligencia artificial para la identificación de personas con acceso al edificio de ingeniería de la Universidad Francisco de Paula Santander Ocaña.		
TITULO EN INGLES	Design of a facial recognition prototype using artificial intelligence techniques to identify people with access to the engineering building of the Francisco de Paula Santander Ocaña University.		
RESUMEN			
Este proyecto pretende generar una nueva alternativa de seguridad a través de la aplicación de diferentes técnicas de inteligencia artificial que permitan elevar el nivel de confiabilidad del edificio de ingenierías de la UFPSO, con el propósito de realizar un seguimiento de la información de acceso de los estamentos universitarios, como: el control de asistencia de los usuarios, el acceso a zonas no autorizadas y la interacción entre la comunidad.			
RESUMEN EN INGLES			
This project aims to generate a new security alternative through the application of different artificial intelligence techniques that allow raising the reliability level of the UFPSO engineering building, with the purpose of monitoring the access information of the estates. university, such as: user attendance control, access to unauthorized areas and interaction between the community.			
PALABRAS CLAVES	Inteligencia Artificial, reconocimiento facial, rostros, acceso, identificación.		
PALABRAS CLAVES EN INGLES	Artificial Intelligence, facial recognition, faces, access, identification.		
CARACTERÍSTICAS			
PÁGINAS: 379	PLANOS: 0	ILUSTRACIONES: 160	CD-ROM: 0



Vía Acolsure, Sede el Algodonal, Ocaña, Colombia - Código postal: 546552
 Línea gratuita nacional: 01 8000 121 022 - PBX: (+57) (7) 569 00 88
 atencionalciudadano@ufps.edu.co - www.ufps.edu.co

Diseño de un prototipo de reconocimiento facial utilizando técnicas de inteligencia artificial para la identificación de personas con acceso al Edificio de Ingeniería de la Universidad Francisco de Paula Santander Ocaña

Juan Pablo Mena Santos
Wilson Javier Arias Cardona

Facultad de Ingenierías, Universidad Francisco de Paula Santander Ocaña
Ingeniería de Sistemas
Msc. Byron Cuesta Quintero
19 de octubre de 2022

Índice

Introducción	15
Capítulo 1. Diseño de un prototipo de reconocimiento facial utilizando técnicas de inteligencia artificial para la identificación de personas con acceso al edificio de ingeniería de la universidad francisco de paula Santander Ocaña.	17
1.1 Planteamiento del problema.....	17
1.2 Formulación del problema	18
1.3 Objetivos	19
1.3.1 Objetivo general	19
1.3.2 Objetivos específicos.....	19
1.4 Justificación.....	20
1.5 Delimitaciones.....	20
1.5.1 Geográfica	20
1.5.2 Temporal	21
1.5.3 Conceptual.....	21
1.5.4 Operativa	21
Capítulo 2. Marco referencial	22
2.1 Marco histórico	22
2.1.1 Antecedentes del reconocimiento facial a nivel internacional	22
2.1.2 Antecedentes del reconocimiento facial a nivel nacional.	26
2.1.3 Antecedentes del reconocimiento facial a nivel local	29
2.2 Marco conceptual	30
2.2.1 Inteligencia artificial (AI).....	31
2.2.2 Machine learning.....	31
2.2.3 Biometría.....	31
2.2.4 Biometría facial	32
2.2.5 Rasgo biométrico.....	32
2.2.6 Phyton.....	32
2.2.7 Qt.....	33
2.2.8 Kivy	33
2.2.9 Tkinter	33
2.2.10 OpenCV	34

2.2.11 NumPy.....	34
2.3 Marco teórico	35
2.3.1 El discriminante de Fisher (LDA).....	35
2.3.2 PCA (Principal Component Analysis)	36
2.3.3 RNA (Red Neuronal Artificial).....	37
2.3.4 RNC (Red Neuronal Convulocional)	37
2.3.5 Perceptrón.....	38
2.3.6 LBPH.....	39
2.4 Marco legal.....	41
2.4.1 Propiedad intelectual	41
2.4.2 Objeto, ámbito de aplicación y definiciones	43
2.4.3 Título II Principios rectores.....	46
2.4.4 Título III Categorías especiales de datos.....	47
2.4.5 Título IV Derechos y condiciones de legalidad para el tratamiento de datos	49
2.4.6 Título V Procedimientos	53
2.4.7 Título VI Deberes de los responsables del tratamiento y encargados del tratamiento	55
2.4.8 Título VII de los mecanismos de vigilancia y sanción. Capítulo I. De la autoridad de protección de datos.....	58
2.4.9 Capítulo II Procedimiento y Sanciones.....	61
2.4.10 Capítulo III del Registro nacional de bases de datos.....	63
2.4.11 Título VIII Transferencia de datos a terceros países.....	64
2.4.12 Título IX Otras disposiciones.....	65
Capítulo 3. Diseño metodológico	67
3.1 Tipo de investigación	67
3.2 Población y muestra	67
3.2.1 Población.....	67
3.2.2 Muestra.....	68
3.3 Estrategia metodológica	68
3.3.1 Modelo de desarrollo de prototipos.....	68
3.4 Técnica e instrumentos de recolección	71
3.5 Análisis de la Información	72
Capítulo 4. Resultados	73
4.1 Estado del arte	73

4.1.1 Biometría.....	73
4.1.2 Historia de la Biometría	74
4.1.3 Reconocimiento Biométrico.....	78
4.1.4 Sistemas de Reconocimiento Biométrico.....	79
4.1.5 Rasgos Biométricos.....	87
4.1.6 Rasgos Biométricos Mayormente Utilizados	88
4.1.7 Reconocimiento de Patrones	109
4.1.8 Técnicas de reconocimiento de patrones.....	117
4.1.9 Sistema de reconocimiento de patrones	181
4.1.10 Reconocimiento facial.....	184
4.1.11 Técnicas de Reconocimiento Facial	192
4.1.12 Técnicas de Reconocimiento Facial Específicas (OpenCV).....	207
4.1.13 Comparativa de las principales técnicas de Reconocimiento Facial.....	221
4.2 Construcción del prototipo funcional de reconocimiento facial	224
4.2.1 Iteración 1	224
4.2.2 Iteración 2.....	262
4.2.3 Iteración 3.....	284
4.2.4 Iteración 4.....	319
4.3 Implementación del sistema.....	342
4.3.1 Iteración 5.....	342
4.3.2 Accesibilidad a la Información.....	352
4.3.3 Escalabilidad	353
4.3.4 Tecnologías Utilizadas y Procedimientos Requeridos	361
Capítulo 5. Conclusiones	367
Capítulo 6. Recomendaciones.....	368
Referencias.....	369

Lista de Tablas

Tabla 1 Bondad de los rasgos biométricos de la cabeza	94
Tabla 2 Bondad de los rasgos biométricos de la mano y de los dedos	100
Tabla 3 Bondad de los rasgos biométricos de todo el cuerpo	104
Tabla 4 Bondad de los rasgos biométricos de comportamiento.....	109
Tabla 5 Características comunes en el reconocimiento de señales	113
Tabla 6 Datos iniciales - Parte I.....	172
Tabla 7 Datos iniciales - Parte II.....	173
Tabla 8 Inicialización aleatoria de centroides.....	173
Tabla 9 Iteración I.....	174
Tabla 10 Elementos del cluster 1 – Iteración I.....	175
Tabla 11 Nuevo centroide 1 – Iteración I.....	175
Tabla 12 Elementos del cluster 2 – Iteración I – Parte I.....	175
Tabla 13 Elementos del cluster 2 – Iteración I – Parte II.....	176
Tabla 14 Nuevo centroide 2 – Iteración I.....	176
Tabla 15 Iteración II.....	176
Tabla 16 Elementos del cluster 1 – Iteración II.....	177
Tabla 17 Nuevo centroide 1 – Iteración II.....	178
Tabla 18 Nuevo centroide 2 – Iteración II.....	178
Tabla 19 Iteración III.....	179
Tabla 20 Elementos del cluster 1 – Iteración III.....	180
Tabla 21 Nuevo centroide 1 – Iteración III.....	180
Tabla 22 Elementos del cluster 3 – Iteración III.....	180
Tabla 23 Nuevo centroide 2 – Iteración III.....	181
Tabla 24 Centroides finales.....	181
Tabla 25 Comparativa entre Visión Humana y Visión Artificial	186
Tabla 26 Aplicaciones de la Visión Artificial.....	187
Tabla 27 Comparativa de las principales técnicas de reconocimiento facial.....	221
Tabla 28 Requerimientos Iniciales del Sistema	225
Tabla 29 Requerimientos funcionales.....	227

Tabla 30	Requerimientos no funcionales.....	229
Tabla 31	Especificaciones técnicas del CPU	232
Tabla 32	Especificaciones técnicas del GPU	234
Tabla 33	Especificaciones técnicas del PSU.....	235
Tabla 34	Especificaciones técnicas de la cámara de videovigilancia	236
Tabla 35	Nivel de satisfacción de los requerimientos iniciales - Iteración 1.....	257
Tabla 36	Nivel de satisfacción de los requerimientos funcionales - Iteración 1.....	258
Tabla 37	Nivel de satisfacción de los requerimientos no funcionales - Iteración 1.....	260
Tabla 38	Anexo de requerimiento no funcional - Iteración 2	263
Tabla 39	Detalle de las tablas del modelo relacional - Iteración 2	272
Tabla 40	Nivel de satisfacción del anexo de requerimiento no funcional - Iteración 2.....	283
Tabla 41	Detalle de las tablas del modelo relacional - Iteración 3	289
Tabla 42	Detalle de las tablas del modelo relacional - Iteración 4	339
Tabla 43	Nivel de satisfacción de los requerimientos iniciales - Iteración 1.....	347
Tabla 44	Nivel de satisfacción de los requerimientos funcionales - Iteración 1.....	348
Tabla 45	Nivel de satisfacción de los requerimientos no funcionales - Iteración 1.....	349
Tabla 46	Herramientas de Software.....	363

Lista de Figuras

Figura 1 Esquema sistema biométrico.....	27
Figura 2 Modelo de prototipos.....	69
Figura 3 Dibujos de Bertillon representativos de la toma de medidas biométricas. Dibujos de Bertillon representativos de la toma de medidas biométricas.....	76
Figura 4 Arquitectura de un sistema de reconocimiento biométrico Arquitectura de un sistema de reconocimiento biométrico.....	79
Figura 5 Operación en modo verificación Operación en modo verificación.....	80
Figura 6 Operación en modo identificación Operación en modo identificación.....	81
Figura 7 Etapas o procesos que componen los sistemas: a. matriculación (enrollment); b. verificación (verification); c. identificación (identification)	85
Figura 8 Reconocimiento de la cara Reconocimiento de la cara.....	89
Figura 9 Termograma facial Termograma facial.....	90
Figura 10 Oreja Oreja	91
Figura 11 (a) Estructura del ojo – (b)Ojo ocultado por el parpado (a) Estructura del ojo – (b)Ojo ocultado por el parpado.....	92
Figura 12 Retina Retina	93
Figura 13 Geometría de la mano Geometría de la mano	95
Figura 14 Geometría de los dedos de la mano Geometría de los dedos de la mano.....	96
Figura 15 (a) Fin de cresta - (b) Bifurcación de cresta (a) Fin de cresta - (b) Bifurcación de cresta	96
Figura 16 (a) Ejemplo de huella - (b) Ejemplo de huella de baja calidad (a) Ejemplo de huella - (b) Ejemplo de huella de baja calidad.....	98
Figura 17 Huella de la mano Huella de la mano.....	98
Figura 18 Venas de la mano y de los dedos Venas de la mano y de los dedos	99
Figura 19 Olor Olor	102
Figura 20 ADN ADN.....	103
Figura 21 Hablador (reconocimiento de voz) Hablador (reconocimiento de voz)	105
Figura 22 Firma Firma.....	106

Figura 23 (a) ejemplo de firma - (b) ejemplo de imitación (a) ejemplo de firma - (b) ejemplo de imitación	107
Figura 24 Forma de andar Forma de andar.....	108
Figura 25 Manera de teclear (Dinámica de tecleo).....	108
Figura 26 Proceso biológico de reconocimiento de patrones para un ser humano Proceso biológico de reconocimiento de patrones para un ser humano	110
Figura 27 Representación comparativa de una red neuronal.....	120
Figura 28 RNA totalmente conectada.....	121
Figura 29 Ejemplo de una neurona con 2 entradas y 1 salida.....	122
Figura 30 Función de activación lineal.....	125
Figura 31 Función de activación sigmoidea	126
Figura 32 Función de activación tangente hiperbólica	127
Figura 33 Arquitectura del perceptrón.....	129
Figura 34 Problema linealmente separable: Función lógica OR	131
Figura 35 Modelo Retro propagación.....	131
Figura 36 Margen m	134
Figura 37 Caso separable linealmente	135
Figura 38 Caso no linealmente separable	137
Figura 39 Muestra parámetros $\xi(i,j)$	140
Figura 40 Árbol de decisión. El espacio es dividido por las superficies de decisión. La superficie que se escoge es una combinación lineal determinada por las funciones de activación.....	148
Figura 41 Proceso determinista: Iluminación del semáforo	151
Figura 42 Esquema del ciclo de diseño de un clasificador	154
Figura 43 Clasificación lineal	156
Figura 44 Clasificador de paralelepípedos.....	159
Figura 45 Representación de la función de coste para un problema de regresión lineal con dos parámetros.....	163
Figura 46 Descenso por gradiente	164
Figura 47 Ejemplo de funcionamiento del ajuste de los parámetros	166
Figura 48 Diferencia entre la función coste de regresión lineal (izquierda) y regresión logística (derecha)	167

Figura 49 Función coste de la regresión logística.....	168
Figura 50 Paso 1: Inicialización aleatoria de los centros.....	169
Figura 51 Paso 2: Asignación de puntos a centros en función de las distancias	170
Figura 52 Paso 3: El centro se reubica pasando a ser el centroide de todos los puntos que tenía asignados.....	171
Figura 53 <i>Algoritmo K-means.</i>	171
Figura 54 Paradigmas del Reconocimiento de patrones.....	181
Figura 55 Esquema de un sistema de reconocimiento de patrones.....	183
Figura 56 Construcción de un modelo 3D con base en un conjunto de imágenes bidimensionales	186
Figura 57 Composición y representación de los canales RGB y Escala de grises de una imagen basada en mapa de bits.....	190
Figura 58 Estructura interna del ojo humano.....	190
Figura 59 Componentes principales de un conjunto de puntos bidimensionales. El primer componente pc1 ofrece una reducción lineal óptima de dimensión de 2D a 1D en cuanto al error cuadrático se refiere	192
Figura 60 Reducción de dimensionalidad.....	193
Figura 61 Autovectores correspondientes a los siete mayores autovalores mostrados como imágenes $p \times p = n$ (Imágenes obtenidas de la BD de rostros ORL).....	194
Figura 62 Imagen promedio de los primeros siete autovectores	195
Figura 63 Autovectores correspondientes a los siete menores autovalores, mostrados como imágenes $p \times p = n$ (Obtenidos de la BD de rostros ORL)	195
Figura 64 Las primeras seis FisherFaces calculadas en la base de datos ORL	197
Figura 65 A diferencia del posicionamiento de datos entre PCA y LPP, este último conserva su estructura local de datos.....	198
Figura 66 Proceso de obtención de parámetros LBPH	201
Figura 67 Identificación de neighborhoods	202
Figura 68 Comparación de una imagen con aplicación de LBP vs imagen artificial.....	203
Figura 69 Esquema de funcionamiento de DCT por bloques.....	205
Figura 70 Alineación de las caras usando DCT por bloques.....	206
Figura 71 Autocaras calculadas sin alinear (superior) y alineadas (inferior)	207

Figura 72 Estructura de clases de los algoritmos de reconocimiento facial de la librería OpenCV	208
Figura 73 Eigenfaces con mapa de colores jet OpenCV	213
Figura 74 Resultado de n Eigenvectores en OpenCV.....	214
Figura 75 Muestra de Fisherfaces	216
Figura 76 Fisherfaces con OpenCV.....	217
Figura 77 Tasa de reconocimiento de rango 1 del método Eigenfaces y Fisherfaces en la base de datos AT&T Facedata.....	219
Figura 78 Operador LBP descrito con una vecindad fija de 3x3	220
Figura 79 Comparativa de núcleos entre el CPU y el GPU.....	233
Figura 80 DS-2CD21893G0-I	239
Figura 81 Top languages over the years - Lenguajes de programación a lo largo de los años .	240
Figura 82 Arquitectura del sistema de reconocimiento facial	244
Figura 83 Cliente de DroidCam – Vista de la cámara.....	247
Figura 84 Arquitectura descriptiva de los procesos del sistema de reconocimiento facial de la iteración 1.....	248
Figura 85 Detección del rostro desde una ventana del SO ejecutando OpenCV.....	249
Figura 86 Ejemplo de fichero con las fotos de los rostros capturados	250
Figura 87 Ejemplo de fotos capturas de tamaño 150 x 150.....	251
Figura 88 Ejemplo de asociación entre el rostro y la etiqueta representativa.....	252
Figura 89 Proceso de detección de personas desconocidas de la iteración 1	253
Figura 90 Matriculación, verificación e identificación de un nuevo usuario en la iteración 1 ..	254
Figura 91 Ejemplo de reconocimiento facial de dos personas con OpenCV.....	255
Figura 92 Arquitectura estructural e integral de las tecnologías utilizadas en la iteración 2	265
Figura 93 Prueba de renderizado de una imagen en la etiqueta canvas, enviada a través de la implementación de WebSockets.....	267
Figura 94 Vídeo transmitido a través del navegador por medio de WebSockets	268
Figura 95 Experimentación de captura y presentación de imagen de vídeo por medio de DroidCam a través del navegador, utilizando UVICORN para la gestión de websockets y Angular	269
Figura 96 Modelo relacional del sistema en la iteración 2.	270

Figura 97 Mockup de la interfaz de presentación y acceso al sistema	274
Figura 98 Mockup de la interfaz de monitoreo.....	274
Figura 99 Mockup de la interfaz de inspección de personas desconocidas.....	275
Figura 100 Mockup de la interfaz de inspección de personas conocidas	277
Figura 101 Mockup de la interfaz de inspección de personas conocidas y desconocidas	278
Figura 102 Mockup de la interfaz detalle de personas conocidas o desconocidas	279
Figura 103 Mockup de la interfaz administrativa para la gestión de usuarios del sistema.....	280
Figura 104 Mockup de la interfaz de registro de usuario	281
Figura 105 Mockup de la interfaz de modificación de registros	282
Figura 106 Funciones de CRUD, validación y notificación.....	286
Figura 107 Ejemplo de visualización de la cámara en el navegador con almacenamiento de información en formularios.....	287
Figura 108 Modelo relacional del sistema en la iteración 3.	287
Figura 109 Interfaz de inicio de sesión.....	290
Figura 110 Interfaz de monitoreo	291
Figura 111 Interfaz de accesos.....	292
Figura 112 Interfaz de validación para el cambio de contraseña del usuario	294
Figura 113 Interfaz de acceso deshabilitado.....	295
Figura 114 Interfaz de búsqueda y asignación de accesos.....	296
Figura 115 Interfaz de módulos	296
Figura 116 Interfaz de modificación de los módulos	298
Figura 117 Interfaz de módulos deshabilitados	298
Figura 118 Interfaz de creación de módulos.....	299
Figura 119 Interfaz de perfiles.....	300
Figura 120 Interfaz de modificación de perfiles.....	302
Figura 121 Interfaz de perfiles deshabilitados.....	303
Figura 122 Interfaz de creación de perfiles	303
Figura 123 Interfaz de personas.....	304
Figura 124 Interfaz de modificación de la persona.....	306
Figura 125 Interfaz de despliegue de la cámara de registro	307
Figura 126 Interfaz de detección y extracción del rostro.....	308

Figura 127 Asociación de etiquetas en el arreglo matricial con Numpy y entrenamiento del modelo.....	308
Figura 128 Interfaz de aprobación del entrenamiento de reconocimiento facial.....	309
Figura 129 Interfaz de desaprobación del entrenamiento de reconocimiento facial	310
Figura 130 Interfaz de creación de personas	311
Figura 131 Interfaz módulo - perfil	312
Figura 132 Interfaz de módulos deshabilitados asociados con el perfil	314
Figura 133 Interfaz de búsqueda de módulos y perfiles para su asociación.....	315
Figura 134 Interfaz de selección del módulo.....	316
Figura 135 Interfaz de selección del módulo, el perfil y mensaje de confirmación	317
Figura 136 Registro de personas conocidas.....	321
Figura 137 Registro de personas desconocidas	323
Figura 138 Evidencia del registro de las personas capturadas.....	323
Figura 139 Filtro de personas reconocidas - Código.....	325
Figura 140 Filtro de personas reconocidas – Nombre de usuario.....	325
Figura 141 Filtro de personas reconocidas – Hora de Registro.....	326
Figura 142 Selección de la fecha	327
Figura 143 Filtro de personas reconocidas – Estado conocido.....	328
Figura 144 Filtro de personas reconocidas – Estado desconocido	329
Figura 145 Interfaz de retroalimentación del entrenamiento del modelo – Progreso de entrenamiento.....	330
Figura 146 Interfaz de retroalimentación del entrenamiento del modelo – Registro completado	331
Figura 147 Interfaz de retroalimentación del entrenamiento del modelo – Cálculo de margen de error.....	332
Figura 148 Interfaz de retroalimentación del entrenamiento del modelo – Registro aprobado	334
Figura 149 Interfaz de retroalimentación del entrenamiento del modelo – Registro reprobado	334
Figura 150 Interfaz del correo electrónico – Creación de cuenta	336
Figura 151 Interfaz del correo electrónico – Actualización de contraseña.....	336
Figura 152 Modelo relacional del sistema en la iteración 4	337
Figura 153 Interfaz de despliegue de múltiples sesiones de transmisión	340

Figura 154 Registro de múltiples personas detectadas	341
Figura 155 Construcción del soporte para la cámara de videovigilancia	343
Figura 156 Prueba de visualización de la cámara de videovigilancia en el laboratorio de redes y telecomunicaciones	344
Figura 157 Base en madera de la cámara de videovigilancia, sujeto al soporte metálico	345
Figura 158 Base de madera pintada de color blanco	346
Figura 159 Sistema NVR.....	355
Figura 160 Arquitectura CCTV y el PSR	355

Capítulo 1. Introducción

Dentro del ambiente organizacional e institucional, garantizar la seguridad de las personas y el acceso a la información sensible es un punto crucial para promover la integridad de las mismas. Un entorno favorable que brinde soluciones efectivas y asegure la información del personal debe implementar diferentes mecanismos, sistemas y recursos con un alto nivel de disponibilidad. En ese sentido y a favor de la seguridad, la biometría se ha convertido en una herramienta importante para el reconocimiento de individuos a través del análisis de características particulares de los seres humanos. Es así como en la antigüedad se utilizaban técnicas de biometría como el registro dactilar para la verificación de la estructura física de las personas; técnicas que con el paso del tiempo se convirtieron en un pilar fundamental para ciencias como la criminología, especialmente en las investigaciones de las personas involucradas en un crimen.

Así mismo, ante el descubrimiento de las ventajas de la biometría, se generó un auge de estudio sobre las técnicas mismas, siendo la biometría facial una de las ramas más abordadas, debido a que es una de las formas naturales más comunes de reconocimiento, que junto con las tecnologías modernas y métodos estadísticos permiten la identificación de personas a través de cámaras de vigilancia, proporcionando valor agregado en los sistemas de seguridad no invasivos de última generación.

Por tal motivo la presente investigación busca el mejoramiento de los procesos organizacionales y el uso de tecnologías de vanguardia de seguridad, a través del desarrollo de

un prototipo funcional de sistema de reconocimiento facial basado en tecnologías de inteligencia artificial como el reconocimiento facial, considerando como entorno de aplicación un sistema de cámara de vigilancia que satisfaga la necesidad de salvaguardar la seguridad e integridad de las personas que accedan al Edificio de Ingeniería de la Universidad Francisco de Paula Santander Ocaña.

Capítulo 2. Diseño de un prototipo de reconocimiento facial utilizando técnicas de inteligencia artificial para la identificación de personas con acceso al edificio de ingeniería de la universidad francisco de paula Santander Ocaña.

2.1 Planteamiento del problema

Actualmente la tecnología se encuentra en constante evolución y con ella el surgimiento de herramientas biométricas con diferentes campos de aplicación. Tal es el caso, que varias empresas buscan implementar instrumentos biométricos que garanticen el desarrollo sostenible dentro de su entorno y procesos organizacionales, con el objetivo de posicionarse como organizaciones líderes en materia tecnológica. Así mismo, hoy en día, es común ver sistemas inteligentes automatizados y altamente complejos comunicándose entre sí, apoyando múltiples áreas en grandes organizaciones para cumplir actividades como: vigilancia, prevención, control de eventos y optimización de procesos. Tareas llevadas a cabo, gracias a la incorporación de paradigmas de aprendizaje de redes neuronales mediante técnicas de “*Machine Learning*”.

No obstante, no solo las organizaciones empresariales han incursionado en el uso de herramientas biométricas, también las Instituciones de Educación Superior han decidido fomentar la investigación y el desarrollo de instrumentos de reconocimiento facial para gestionar el control de acceso en áreas de vital importancia, ya que la inseguridad es un problema de carácter urgente que se suscita en toda organización.

En ese sentido, la Universidad Francisco de Paula Santander seccional Ocaña no ha sido ajena a estas fallas de seguridad; anteriormente, existían áreas sensibles, expuestas a vulnerabilidades, donde se carecía de verificación en el control y vigilancia del personal, como en la antigua oficina de la Facultad de Ingenierías de la UFPSO, ubicada en la sección de anexos, donde no se disponía de un espacio amplio de trabajo en el que laborar, razón por la que tanto estudiantes y personal administrativo contaban con acceso a información confidencial, pudiendo llevarse a cabo afectaciones a la integridad de los usuarios y de la información, bajo modalidades de: suplantación de identidad, robo y alteración de documentos. Tal situación logró contrarrestarse con la construcción del Edificio de Ingeniería en el año 2019, materializando un filtro de acceso a áreas administrativas con la intención de restringir el ingreso no autorizado especialmente del estamento estudiantil y por ende salvaguardar la información. Sin embargo, el proceso de identificación del personal sigue siendo una falla de seguridad en la mayoría de las dependencias, debido a que no existen alternativas o estrategias de fortalecimiento en cuanto a seguridad administrativa.

De igual forma, al considerar que la universidad enmarca en su misión institucional la búsqueda de estrategias de innovación y el uso de tecnologías, resulta indispensable que su forma de abordar el concepto de seguridad comience a generar nuevas alternativas de confianza que le permitan posicionarse a la vanguardia de la tecnología en el panorama nacional e internacional.

2.2 Formulación del problema

¿Con el diseño de un prototipo de reconocimiento facial utilizando técnicas de inteligencia artificial para la identificación de personas con acceso al Edificio de Ingeniería de la Universidad Francisco de Paula Santander seccional Ocaña, se podrá afianzar el uso de herramientas tecnológicas de reconocimiento de patrones como técnicas para la seguridad y vigilancia del mismo?

2.3 Objetivos

2.3.1 Objetivo general

Diseñar un prototipo de reconocimiento facial utilizando técnicas de inteligencia artificial para la identificación de personas con acceso al Edificio de Ingeniería de la Universidad Francisco de Paula Santander seccional Ocaña.

2.3.2 Objetivos específicos

- Elaborar un estado del arte sobre las diferentes herramientas y técnicas para el reconocimiento de patrones.
- Construir un prototipo funcional de reconocimiento facial de acuerdo con las herramientas y técnicas establecidas.
- Validar el correcto funcionamiento del prototipo de reconocimiento facial en un ambiente real.

2.4 Justificación

El presente proyecto tiene como propósito generar una nueva alternativa de seguridad que fomente la investigación, desarrollo e implementación experimental de un prototipo de sistema de seguridad utilizando técnicas de reconocimiento facial para la identificación de personas del Edificio de Ingeniería que permita elevar el nivel confiabilidad del establecimiento, donde sea posible reconocer las características principales de los rostros y compararlos con una base de datos centralizada suministrada por la universidad, para efectuar el reconocimiento.

Adicionalmente con este proyecto se podrá realizar una trazabilidad de la información de acceso, que permitirá abrir el camino para diversos estudios de aplicabilidad de inteligencia artificial y biometría facial, abarcando diferentes soluciones como: el control de asistencia de los diferentes estamentos universitarios, el acceso a zonas no autorizadas, la interacción entre la comunidad y la prevención de situaciones de riesgo; sumado a lo anterior, la UFPSO podría posicionarse como una universidad precursora en el uso de tecnología avanzada para el fortalecimiento de sus procesos organizacionales, consolidando así su paso hacia la acreditación.

2.5 Delimitaciones

2.5.1 Geográfica

El siguiente proyecto se llevará a cabo en el Edificio de Ingeniería de la Universidad Francisco de Paula Santander seccional Ocaña.

2.5.2 Temporal

El siguiente proyecto se llevará a cabo en 12 semanas, fundamentados en las actividades a desarrollar.

2.5.3 Conceptual

Aplicando conocimientos en Base de datos, Lenguajes de Programación (Python o C++), Biometría Facial, Inteligencia Artificial (AI), Análisis de Componentes Principales (PCA).

2.5.4 Operativa

Este proyecto estará determinado por una metodología de desarrollo de prototipos para la agilización en la construcción del prototipo final.

Capítulo 3. Marco referencial

3.1 Marco histórico

3.1.1 Antecedentes del reconocimiento facial a nivel internacional

A nivel internacional se ha investigado y desarrollado múltiples sistemas de reconocimiento facial con diferentes funcionalidades, enfocados en aplicaciones para el control de acceso a personas en diferentes áreas dentro de zonas no autorizadas o restringidas.

3.1.1.1 Análisis y diseño de un sistema de reconocimiento facial aplicando machine learning para detectar e identificar intrusos. Es así como es importante resaltar el trabajo investigativo desarrollado por Portilla (2018), trabajo que tenía como objetivo realizar un análisis y un diseño de un sistema de reconocimiento facial aplicando técnicas de “Machine Learning” para detectar e identificar intrusos.

Dicho objetivo se logró a través de tareas tales como: la determinación de las mejores herramientas tecnológicas para su utilización en la validación de la información recolectada por la cámara de vídeo, el diseño del modelo entidad relación para el almacenamiento de los datos del sistema, el desarrollo del programa por medio del lenguaje de programación Python para el reconocimiento facial a partir de las imágenes captadas en vídeo y el desarrollo total de la aplicación con herramientas web, llevando el registro de las personas que contaban con acceso al

edificio de las Carrera de Ingeniería en Sistemas Computacionales e Ingeniería en Networking y Telecomunicaciones de la Universidad de Guayaquil.

3.1.1.2 Una aplicación de reconocimiento facial para el cuidado de personas mayores (A facial recognition application for elderly care). De igual forma el proyecto presentado por Martikainen y Said, titulado: “Una Aplicación de Reconocimiento Facial para el Cuidado de Adultos Mayores” (2018), refleja el interés que se ha llevado en las últimas décadas sobre el reconocimiento facial, debido al potencial computacional con la que se cuenta actualmente y el acceso a enormes cantidades de datos, facilitando la aplicabilidad de dicha tecnología.

Martikainen y Said (2018) manifiestan que el reconocimiento facial es una de las aplicaciones más habituales de biometría y que la población sueca cuenta con un alto porcentaje de adultos mayores que por lo general viven aislados o en total soledad, generando como desafío dentro de su sociedad apoyar la autonomía y bienestar de los adultos mayores.

Por tal motivo, Martikainen y Said (2018) en su tesis, presentan un estudio del potencial que tiene el reconocimiento facial en áreas de monitoreo y cuidado de adultos mayores, por medio de la presentación e implementación de un prototipo de una arquitectura de un sistema de reconocimiento facial, con la finalidad de evaluar su viabilidad.

De esta forma obtuvieron resultados importantes, en los que lograron validar la realidad de dicha necesidad y que un sistema de reconocimiento facial potencializado podría solventarla,

ya que su sistema lograba determinar el reconocimiento facial de las personas mayores, pero era inconsistente.

3.1.1.3 Diseño de un sistema prototipo de reconocimiento facial para la identificación de personas en la facultad de ingeniería en ciencias aplicadas (FICA) de la universidad técnica del norte utilizando técnicas de inteligencia artificial. El proyecto titulado: “Diseño de un Sistema Prototipo de Reconocimiento Facial para la Identificación de Personas en la Facultad de Ingeniería en Ciencias Aplicadas (FICA) de la Universidad Técnica del Norte Utilizando Técnicas de Inteligencia Artificial”, presentado por Bolívar (2019), manifiesta la importancia y la multiplicidad de aplicaciones de monitoreo, control de acceso y seguridad que incorporan técnicas de reconocimiento facial, resaltando el hecho de que este tipo de técnica de biometría destaca sobre las demás, debido a que las personas pueden ser reconocidas a distancia, permitiendo incorporar sistemas de reconocimiento facial en instituciones con la finalidad de restringir el acceso a personal no autorizado, evitando afectaciones al bien público y privado.

Por esta razón, el objetivo de su trabajo investigativo fue identificar personas en entornos controlados y no controlados dentro del edificio universitario de la FICA, debido a que ha sufrido afectaciones a la seguridad institucional que han generado problemáticas de inseguridad, motivo que los llevó a plantear un diseño de un sistema completo de reconocimiento facial donde combinaron técnicas de Inteligencia Artificial, tales como las arquitecturas de redes neuronales convolucionales, el algoritmo de máquinas de vector soporte y el procesamiento paralelo, bajo una estructura de procesamiento gráfico por medio de una GPU.

Su sistema procedía con el entrenamiento de la red neuronal utilizando un conjunto de datos VGGFace2, con la finalidad de generar el aprendizaje de la red y la generalización de rasgos faciales con un tamaño de 512 bytes por rostro, apoyados de la supervisión conjunta de las señales de pérdida de softmax y la pérdida central.

Siendo así, como a través de un clasificador lograron desarrollar varios experimentos de diferentes clases, obteniendo como resultado una eficiencia en respuesta del reconocimiento facial en tiempo real a partir de una muestra de individuos experimentales para el entrenamiento del sistema, resultando en una respuesta favorable para establecer un punto de partida en el desarrollo de este tipo de sistemas biométricos en múltiples entornos.

3.1.1.4 Reconocimiento facial a distancia para un sistema de control de acceso

independiente (Face recognition at a distance for a stand-alone access control system). La

siguiente investigación realizada por Lee, Park, Yoo, Jung y Huh (2019), titulada:

“Reconocimiento Facial a Distancia para un Sistema de Control de Acceso Independiente”,

manifiesta el hecho de que los sistemas de control basados en el reconocimiento de rostros han tomado gran popularidad en aplicaciones de consumo, pero a pesar de esto, la implementación de un sistema de control de acceso independiente aún se enfrentaría a varios problemas e inconvenientes, debido a que en muchas ocasiones la falta de recursos computacionales requieren de algoritmos eficientes y complejos, pero con altos niveles de ligereza.

A su vez, los autores resaltan que los sistemas de control convencionales requieren de una cooperación activa bastante significativa, independiente de su naturaleza de servicio, además

los sistemas de reconocimiento facial se ven afectados por agentes externos como los cambios de iluminación, siendo uno de los problemas de alta complejidad dentro de las aplicaciones de control de acceso de este tipo.

Es así como los autores traen a acotación el diseño y la implementación de un sistema de control de acceso práctico y autónomo, basado en el reconocimiento de rostros humanos a distancia, aplicando LBP Adaboost para la detección de rostros y ojos, debido a su rapidez e invariabilidad a los cambios de iluminación.

El sistema puede detectar rostros y ojos de distintos tamaños a una distancia bastante extensa. Para lograr un sistema eficiente y eficaz, realizaron modificaciones como la sustitución de la ondícula de Gabor con filtrado derivado de Gauss, reduciendo el tamaño del rasgo facial en un 40% de los rasgos faciales basados de Gabor-LBP, garantizando robustez frente a los cambios de iluminación mencionados anteriormente.

Como resultados de la experimentación, se obtuvieron cifras de un 97.27% en referencias de precisión de reconocimiento facial en conjuntos de datos E-face y un 99.06% en conjuntos de datos XM2VTS. Permitiendo así una tasa de aceptación real de 91.5%, una tasa de aceptación falsa de 0.28% y un promedio de velocidad de procesamiento de 5.26 fps en un conjunto de datos de video e imagen recién recopilados en un entorno de oficina.

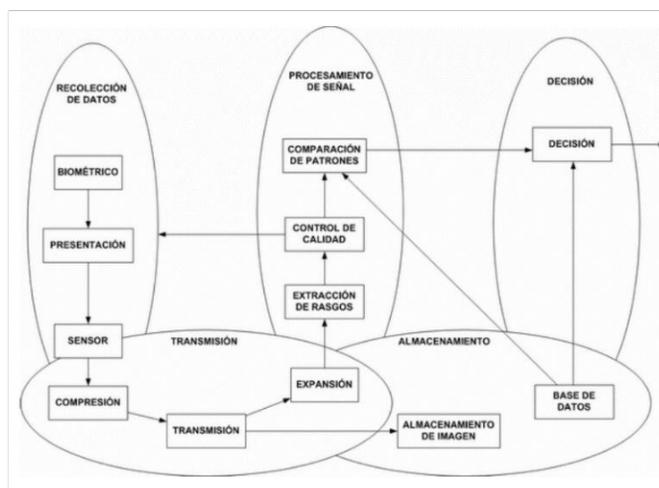
3.1.2 Antecedentes del reconocimiento facial a nivel nacional

El ambiente investigativo sobre la temática a nivel nacional ha presentado una adaptación por parte de los investigadores, donde se han tomado en cuenta temas de estudio no solo con el reconocimiento facial sino con el contexto global que rodea a la temática de la biometría y sus diferentes técnicas para ser implementadas en sistemas digitales automatizados, supervisados y no supervisados.

3.1.2.1 Técnicas de biometría basadas en patrones faciales del ser humano. López y Toro (2012) a través de su monografía titulada: “Técnicas de Biometría Basadas en Patrones Faciales del Ser Humano”, presentan dentro de su proceso investigativo un estado del arte que abarca toda la temática de los sistemas biométricos tanto las técnicas del reconocimiento de patrones.

Figura 1 Esquema sistema biométrico.

Esquema sistema biométrico.



Nota. Datos tomados de (Nicolas & Toro, 2012)

Siendo de esta forma como resaltan la necesidad de proteger los recursos empresariales a la par de la evolución del mundo por medio de la biometría informática, debido a que permite la restricción de acceso a recursos sensibles.

A su vez, López y Toro (2012) resaltan la importancia que representa el reconocimiento biométrico, no solo en áreas de control de acceso sino en campos como la criminología y las ciencias forenses, debido a su nivel de complejidad, ya que facilitaría el proceso investigativo de los casos criminológicos.

Es importante resaltar que López y Toro (2012) realizaron un proceso investigativo profundo donde no solo hablan de las técnicas biométricas a nivel fisiológico sino también a nivel biológico, donde abarcan temas tales como la identificación de personas a través de la estructura del ácido desoxirribonucleico, siendo esta una forma de reconocimiento de personas con un alto nivel de credibilidad.

3.1.2.2 Sistema de reconocimiento facial basado en imágenes con color. Es así, como desarrollaron un sistema de reconocimiento facial constituido por fases donde en primera instancia se detectaba y localizaba el rostro humano por medio, aplicando el algoritmo de aprendizaje AdaBoost y clasificadores en cascada, con el fin de amplificar la efectividad de la detección. En el mismo orden de ideas, dentro de su segunda fase aplicaron el método de eigenfaces e implementaron un sistema que les permitió el reconocimiento y la identificación del rostro de la persona de entrada con una persona específica registrada en el sistema, todo esto bajo técnicas de distancia euclidiana y de mahalanobis.

Finalmente, como resultados pudieron comparar y determinar que el sistema propuesto tanto en escala de grises como en imágenes a color, en contraste con la información del color en el plano HSV, si puede mejorar el proceso de reconocimiento teniendo en cuenta el punto de comparación con un plano RGB. (Pedraza Pico, Rondón, & Arguello, 2011)

3.1.3 Antecedentes del reconocimiento facial a nivel local

En el ámbito local el estudio y la implementación de tecnologías de reconocimiento facial se encuentra exento de su aplicabilidad y estudio, motivo por el que la Universidad Francisco de Paula Santander seccional Ocaña, sirve a la comunidad educativa como pilar investigativo para el fomento y desarrollo de aplicaciones que incorporen estas tecnologías de vanguardia. A pesar de dicha problemática, es destacado mencionar un trabajo investigativo local que se fundamenta por medio de un estado del arte completo, el cual se presenta a continuación.

3.1.3.1 Prototipo de un sistema de reconocimiento facial para ingreso a la biblioteca de la UFPSO. Dentro de los antecedes a nivel local, la investigación en áreas como el reconocimiento facial es escasa, pero se logra rescatar y resaltar el trabajo investigativo presentado por Uribe y García (2018), titulado como: “Prototipo de un Sistema de Reconocimiento Facial Para Ingreso a la Biblioteca de la UFPSO”, dicho trabajo investigativo resalta el avance tecnológico actual en áreas del reconocimiento facial, lo que ha generado un alza en el uso e investigación de estas herramientas tecnológicas, debido a su aplicabilidad en multiplicidad de actividades cotidianas.

Es así como los autores enfatizan que una de las formas de identificación de personas comúnmente utilizada es el reconocimiento del rostro, debido a que es uno de los rasgos indistinguible y característicos de las personas. A su vez, estudios como la inteligencia artificial, reconocimiento de señales y reconocimiento de patrones ha tenido gran acogida dentro del campo investigativo, gracias a su nivel de complejidad.

Dentro de los procesos que abarca el reconocimiento facial, se requiere la ubicación de un conjunto de caras comprendidos en una imagen, donde se analizan las características y rasgos principales del rostro de las personas, siendo comparados dentro de un conjunto de datos centralizados y almacenados en una base de datos.

De esta forma, Uribe y García (2018), llevan a cabo el trabajo de estudio y diseño de un prototipo de sistema automático de reconocimiento facial con la finalidad de controlar el acceso de ingreso a la biblioteca Argemiro Bayona Portillo de la Universidad Francisco de Paula Santander Ocaña, a través de la descripción y análisis de las técnicas de reconocimiento facial de mayor implementación, utilizadas en los últimos años.

Optando para el desarrollo del prototipo la técnica de Análisis de Componentes Principales, debido a que es fiable en sus resultados, complejidad y rapidez de ejecución, garantizando el proceso de reconocimiento facial.

3.2 Marco conceptual

3.2.1 Inteligencia artificial (AI)

“Es la ciencia de construir máquinas para que hagan cosas que, si las hicieran los humanos, requerirían inteligencia. Es aquella ciencia que incorpora conocimiento a los procesos o actividades para que estos tengan éxito”. (Alfonso, Cazorla, Colomina, Escolano, & Lozano, 2003)

3.2.2 Machine learning

“The field of machine learning does rely upon nature's bounty for both inspiration and mechanism. Many machine learning systems now borrow heavily from current thinking in cognitive science, and rekindled interest in neural networks and connectionism is evidence of serious mechanistic and philosophical currents running through the field”. [El campo del aprendizaje automático se basa en la generosidad de la naturaleza tanto para la inspiración como para el mecanismo. Muchos sistemas de aprendizaje automático ahora toman prestados en gran medida del pensamiento actual en ciencia cognitiva, y el interés reavivado en las redes neuronales y el conexionismo es evidencia de serias corrientes mecanicistas y filosóficas que atraviesan el campo]. (Goldberg & Holland, 1988)

3.2.3 Biometría

“La biometría se define como la ciencia que establece la identidad de un individuo basándose en los atributos tanto físicos, como químicos y de comportamiento de una persona”. (Pérez, 2017)

3.2.4 Biometría facial

“Es aquella que utiliza la información que se encuentra en el rostro del individuo con el fin de identificarlo de manera automática”. (Pérez, 2017)

“Las técnicas biométricas (del griego bios = vida, y metrón = medida) permiten la identificación de las personas basándose en el reconocimiento de sus peculiaridades físicas, propias e individuales”. (Caldera & Zapico, 2009)

3.2.5 Rasgo biométrico

“Un rasgo biométrico es una característica humana que se puede medir con el fin de conocer la identidad del usuario.” (Pérez, 2017)

3.2.6 Python

“Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python’s elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language

for scripting and rapid application development in many areas on most platforms.” [Python es un potente lenguaje de programación fácil de aprender. Tiene estructuras de datos eficientes de alto nivel y un enfoque simple pero efectivo para la programación orientada a objetos. La elegante sintaxis y la escritura dinámica de Python, junto con su naturaleza interpretada, lo convierten en un lenguaje ideal para la creación de scripts y el rápido desarrollo de aplicaciones en muchas áreas en la mayoría de las plataformas]. (van Rossum, 2020)

3.2.7 Qt

“Qt for Python is the project that provides the official set of Python bindings (PySide2) that will supercharge your Python applications.” [Qt para Python es el proyecto que proporciona el conjunto oficial de enlaces de Python (PySide2) que potenciarán sus aplicaciones de Python]. (The QT Company, 2020)

3.2.8 Kivy

“Open-source Python library for rapid development of applications that make use of innovative user interfaces, such as multi-touch apps.” [Biblioteca de Python de código abierto para el desarrollo rápido de aplicaciones que utilizan interfaces de usuario innovadoras, como aplicaciones multitáctiles]. (Kivy, 2021)

3.2.9 Tkinter

“The tkinter package (“Tk interface”) is the standard Python interface to the Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, as well as on Windows systems. (Tk itself is not part of Python; it is maintained at ActiveState.)” [El paquete tkinter (“interfaz Tk”) es la interfaz estándar de Python para el kit de herramientas Tk GUI. Tanto Tk como tkinter están disponibles en la mayoría de las plataformas Unix, así como en los sistemas Windows. (Tk en sí no es parte de Python; se mantiene en ActiveState)]. (Tkiter, 2021)

3.2.10 OpenCV

“OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.” [OpenCV (Open Source Computer Vision Library) es una biblioteca de software de visión artificial y aprendizaje automático de código abierto. OpenCV se creó para proporcionar una infraestructura común para aplicaciones de visión por computadora y para acelerar el uso de la percepción de la máquina en los productos comerciales. Al ser un producto con licencia BSD, OpenCV facilita que las empresas utilicen y modifiquen el código]. (OpenCV, 2021)

3.2.11 NumPy

“NumPy is an open-source project aiming to enable numerical computing with Python. It was created in 2005, building on the early work of the Numerical and Numarray libraries. NumPy will always be 100% open-source software, free for all to use and released under the liberal terms of the modified BSD license.” [NumPy es un proyecto de código abierto que tiene como objetivo permitir la computación numérica con Python. Fue creado en 2005, basándose en el trabajo inicial de las bibliotecas Numerical y Numarray. NumPy siempre será un software 100% de código abierto, de uso gratuito para todos y publicado bajo los términos liberales de la licencia BSD modificada].

3.3 Marco teórico

3.3.1 El discriminante de Fisher (LDA)

“El Análisis Discriminante Lineal o Linear Discriminant Analysis (LDA) es un método de clasificación supervisado de variables cualitativas en el que dos o más grupos son conocidos a priori y nuevas observaciones se clasifican en uno de ellos en función de sus características. Haciendo uso del teorema de Bayes, LDA estima la probabilidad de que una observación, dado un determinado valor de los predictores, pertenezca a cada una de las clases de la variable cualitativa.” (Amat R. , 2020)

“A análise discriminante, também conhecida como discriminante linear de Fisher (FLDA), é uma técnica que está se tornando muito comum em aplicações de visão computacional. Essa técnica utiliza informações das categorias associadas a cada padrão para

extrair linearmente as características mais discriminantes. Através da análise discriminante de Fisher podemos também realizar a discriminação entre classes, através de processos supervisionados (quando se conhece o padrão) ou através de processos não supervisionados, em que é utilizado quando não se tem um padrão conhecido”. [El análisis discriminante, también conocido como discriminante lineal de Fisher (FLDA), es una técnica que se está volviendo muy común en las aplicaciones de visión por computadora. Esta técnica utiliza información de las categorías asociadas con cada patrón para extraer linealmente las características más discriminatorias. Mediante el análisis discriminante de Fisher también podemos discriminar entre clases, mediante procesos supervisados (cuando se conoce el patrón) o mediante procesos no supervisados, donde se utiliza cuando no hay patrón conocido]. (Amorim & Pistori, 2007)

3.3.2 PCA (Principal Component Analysis)

“Principal Component Analysis, or PCA, is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set”. [El análisis de componentes principales, o PCA, es un método de reducción de dimensionalidad que se utiliza a menudo para reducir la dimensionalidad de grandes conjuntos de datos, transformando un gran conjunto de variables en uno más pequeño que aún contiene la mayor parte de la información en el conjunto grande]. (Zakaria, 2020)

“Principal component analysis (PCA) is a multivariate technique that analyzes a data table in which observations are described by several inter-correlated quantitative dependent

variables. Its goal is to extract the important information from the table, to represent it as a set of new orthogonal variables called principal components, and to display the pattern of similarity of the observations and of the variables as points in maps. The quality of the PCA model can be evaluated using cross-validation techniques such as the bootstrap and the jackknife”. [El análisis de componentes principales (PCA) es una técnica multivariante que analiza una tabla de datos en la que las observaciones se describen mediante varias variables dependientes cuantitativas interrelacionadas. Su objetivo es extraer la información importante de la tabla, representarla como un conjunto de nuevas variables ortogonales llamadas componentes principales y mostrar el patrón de similitud de las observaciones y de las variables como puntos en mapas. La calidad del modelo de PCA se puede evaluar mediante técnicas de validación cruzada como el bootstrap y el jackknife]. (Abdi & Williams, 2010)

3.3.3 RNA (Red Neuronal Artificial)

“Una red neuronal artificial es un procesador distribuido en paralelo de forma masiva que tiene una tendencia natural para almacenar conocimiento de forma experimental y lo hace disponible para su uso.” (Plata, 2020)

3.3.4 RNC (Red Neuronal Convulocional)

Actualmente dentro de trabajos investigativos se hace atractivo el estudio de temas relacionados con el Deep Learning como son las redes convolucionales, debido a que garantizan

un desempeño frente a problemas de reconocimiento, análisis e interpretación de objetos en imágenes y vídeos.

Su funcionalidad parte de la extracción de características dentro de una imagen, con el fin de ser utilizadas en la clasificación de objetos, esto se logra utilizan filtros entre capas que se ajustarán y optimizarán con los componentes de clasificación, garantizando reducir el margen de error de la clasificación general. (Massiris, Delrieux, & Fernández Muñoz, 2018)

3.3.5 Perceptrón

“El perceptrón es la forma más simple de una red neuronal usada para la clasificación de un tipo especial de patrones, los linealmente separables (es decir, patrones que se encuentran a ambos lados de un hiperplano). Básicamente, consiste de una neurona con pesos sinápticos y umbral ajustables.” (Anguas Pérez, 2020)

“El perceptrón es una red de alimentación directa, esto es la información fluye desde la capa de entrada hacia la capa de salida.

El Perceptrón es un clasificador, asigna a un vector de N valores un valor binario, usando una transformación no lineal. Así cada vector pertenece a una de las particiones que crea el perceptrón.

El perceptrón es una máquina de computación universal y tiene la expresividad equivalente a la lógica binaria ya que podemos crear un perceptrón que tenga el mismo comportamiento que una función booleana NAND y a partir de esta función se puede crear cualquier otra función booleana.” (Neural Networks Framework, 2020)

“El objetivo del Perceptrón es encontrar un hiperplano capaz de separar correctamente un conjunto de datos que sean linealmente separables, una vez obtenido el hiperplano, este puede utilizarse para clasificaciones binarias. Aunque el Perceptrón es un algoritmo de aprendizaje muy simple, entender su funcionamiento es clave para aprender otros métodos más complejos como las máquinas de vector soporte SVM o las redes neuronales artificiales.” (Amat J. , 2018)

3.3.6 LBPH

Replicar la capacidad de reconocimiento de rostros en sistemas computacionales, igualando el nivel de complejidad que posee el cerebro humano para dicha tarea, es una actividad que requiere de algoritmos de análisis de imágenes de alta complejidad, debido a la desafiante variabilidad que existe dentro de las expresiones faciales, apariencias y variantes de iluminación. Es así como se hace necesario, para lograr un sistema de reconocimiento facial en tiempo real con una alta precisión, reducir la dimensionalidad de la imagen, permitiendo una mejor clasificación, teniendo en cuenta el volumen de procesamiento computacional que se requiere.

“LBPH is one of the most popular conventional methods; it is used for robust data representation, as well as histograms, for features reduction.” [LBPH es uno de los métodos

convencionales más populares; Se utiliza para una representación robusta de datos, así como histogramas, para la reducción de características]. (Abuzneid & Mahmood, 2018)

Los sistemas de reconocimiento facial están determinados por tres fases principales consecutivas: Preprocesamiento, Extracción de Características y Clasificación. La fase de Preprocesamiento es el primer paso, está conformado por múltiples tipos de operaciones, como: la captura y registro de la imagen, el escalado, la normalización, la reducción del ruido de fondo, la detección del rostro y el redimensionamiento de la imagen para conservar las características importantes, facilitando el procesamiento complejo.

Siguiendo este mismo orden de ideas, dentro de la segunda fase, la fase de Extracción de Características, se realiza un mapeo de la imagen por medio de la codificación binaria, según la diferencia de píxeles centrales con sus vecinos dentro de una matriz preestablecida de ocho píxeles, permitiendo generar histogramas concatenados de dichos códigos binarios que garanticen la extracción de características, reduciendo el ruido contenido en la imagen debido a la variabilidad de iluminación.

Es así, como para su tercera fase se realiza una clasificación de las características extraídas de la imagen que permiten la detección de la persona que está siendo analizada en la imagen.

“LBP outperformed most of the global extraction feature methods such as Principal Component Analysis (PCA), independent component analysis (ICA) and Linear Discriminate

Analysis (LDA), in addition to the PCA-inherited methods such as Diagonal PCA, Curvelet-based PCA, Kernel PCA, 2-DPCA and Kernel FLD.” [LBP superó a la mayoría de los métodos de características de extracción global como el análisis de componentes principales (PCA), el análisis de componentes independientes (ICA) y el análisis de discriminación lineal (LDA), además de los métodos heredados de PCA como Diagonal PCA, PCA basado en Curvelet, Kernel PCA, 2-DPCA y Kernel FLD]. (Abuzneid & Mahmood, 2018)

3.4 Marco legal

3.4.1 Propiedad intelectual

Según Mineducación (2009), “En Colombia la Propiedad Intelectual se define como aquella que se ejerce sobre las creaciones intelectuales, producto del talento humano y que constituyen en sí mismas bienes de carácter inmaterial, objeto de protección a través de diferentes normas jurídicas. Las creaciones intelectuales que son objeto de la propiedad intelectual versan sobre dos concepciones diferentes. Una de ellas, referida a la estética, específicamente las obras literarias y las obras artísticas, corresponde al derecho de autor; y las otras, referidas a la actividad industrial, como las marcas y las patentes, se ubican en la propiedad industrial”, (Morata, 2011).

Apps.co (2015), describe 10 aspectos claves sobre derechos de autor para crear una App de los cuales se mencionan los puntos relevantes en materia de legislación nacional referente al tema:

3.4.1.1 Normas relevantes que rigen la propiedad intelectual del software en Colombia. A

continuación, se enuncian las normas relevantes en Colombia en materia de propiedad intelectual.

- Decisión Andina 486 de 2000
- Decisión Andina 351 de 1993
- Ley 23 de 1982
- Ley 1450 del 16 de junio de 2011

3.4.1.2 Derechos morales. Son aquellos que perduran eternamente en el tiempo, el más relevante es el de Paternidad. El autor siempre debe ser reconocido como creador de una obra a menos que decida lo contrario. Por tal razón, en Colombia es recomendable incluir el nombre de los desarrolladores en los términos de uso del software.

3.4.1.3 Derechos patrimoniales. Son derechos transferibles que involucran los posibles usos de una obra: goce, reproducción, comunicación pública, puesta a disposición del público, distribución.

3.4.1.4 Licencias del software. Las licencias son autorizaciones de uso del software y/o el código. Permiten usar un software o su código bajo ciertas condiciones, más no transfiere los derechos. Quien tiene los derechos patrimoniales, es decir el titular es quien emite las licencias.

Depende del modelo de negocio se deben ver cuáles de estas licencias se adaptan mejor a las necesidades.

3.4.1.5 Trámites importantes para la protección del software. Dentro de los trámites a realizar para la protección de software se encuentran, el registro de código ante Dirección Nacional de Derecho de Autor y Software, registro de acuerdos sobre la titularidad el software y el registro de las licencias de uso sobre el programa.

3.4.2 Objeto, ámbito de aplicación y definiciones

3.4.2.1 Artículo 1º. Objeto. La presente ley tiene por objeto desarrollar el derecho constitucional que tienen todas las personas a conocer, actualizar y rectificar las informaciones que se hayan recogido sobre ellas en bases de datos o archivos, y los demás derechos, libertades y garantías constitucionales a que se refiere el artículo 15 de la Constitución Política; así como el derecho a la información consagrado en el artículo 20 de la misma.

3.4.2.2 Artículo 2º. Ámbito de aplicación. Los principios y disposiciones contenidas en la presente ley serán aplicables a los datos personales registrados en cualquier base de datos que los haga susceptibles de tratamiento por entidades de naturaleza pública o privada.

La presente ley aplicará al tratamiento de datos personales efectuado en territorio colombiano o cuando al responsable del Tratamiento o Encargado del Tratamiento no

establecido en territorio nacional le sea aplicable la legislación colombiana en virtud de normas y tratados internacionales.

El régimen de protección de datos personales que se establece en la presente ley no será de aplicación:

- a) A las bases de datos o archivos mantenidos en un ámbito exclusivamente personal o doméstico.
- b) Cuando estas bases de datos o archivos vayan a ser suministrados a terceros se deberá, de manera previa, informar al Titular y solicitar su autorización. En este caso los responsables y Encargados de las bases de datos y archivos quedarán sujetos a las disposiciones contenidas en la presente ley.
- c) A las bases de datos y archivos que tengan por finalidad la seguridad y defensa nacional, así como la prevención, detección, monitoreo y control del lavado de activos y el financiamiento del terrorismo.
- d) A las Bases de datos que tengan como fin y contengan información de inteligencia y contrainteligencia.
- e) A las bases de datos y archivos de información periodística y otros contenidos editoriales; e) A las bases de datos y archivos regulados por la Ley 1266 de 2008.
- f) A las bases de datos y archivos regulados por la Ley 79 de 1993.

Parágrafo. Los principios sobre protección de datos serán aplicables a todas las bases de datos, incluidas las exceptuadas en el presente artículo, con los límites dispuestos en la presente

ley y sin reñir con los datos que tienen características de estar amparados por la reserva legal. En el evento que la normatividad especial que regule las bases de datos exceptuadas prevea principios que tengan en consideración la naturaleza especial de datos, los mismos aplicarán de manera concurrente a los previstos en la presente ley.

3.4.2.3 Artículo 3°. Definiciones. Para los efectos de la presente ley, se entiende por:

- a) **Autorización:** Consentimiento previo, expreso e informado del Titular para llevar a cabo el Tratamiento de datos personales.
- b) **Base de Datos:** Conjunto organizado de datos personales que sea objeto de Tratamiento.
- c) **Dato personal:** Cualquier información vinculada o que pueda asociarse a una o varias personas naturales determinadas o determinables.
- d) **Encargado del Tratamiento:** Persona natural o jurídica, pública o privada, que por sí misma o en asocio con otros, realice el Tratamiento de datos personales por cuenta del responsable del Tratamiento.
- e) **Responsable del Tratamiento:** Persona natural o jurídica, pública o privada, que por sí misma o en asocio con otros, decida sobre la base de datos y/o el Tratamiento de los datos.
- f) **Titular:** Persona natural cuyos datos personales sean objeto de Tratamiento.
- g) **Tratamiento:** Cualquier operación o conjunto de operaciones sobre datos personales, tales como la recolección, almacenamiento, uso, circulación o supresión.

3.4.3 Título II Principios rectores

3.4.3.1 Artículo 4º. Principios para el tratamiento de datos personales. En el desarrollo, interpretación y aplicación de la presente ley, se aplicarán, de manera armónica e integral, los siguientes principios:

- a) **Principio de legalidad en materia de Tratamiento de datos:** El Tratamiento a que se refiere la presente ley es una actividad reglada que debe sujetarse a lo establecido en ella y en las demás disposiciones que la desarrollen.
- b) **Principio de finalidad:** El Tratamiento debe obedecer a una finalidad legítima de acuerdo con la Constitución y la Ley, la cual debe ser informada al Titular.
- c) **Principio de libertad:** El Tratamiento sólo puede ejercerse con el consentimiento, previo, expreso e informado del Titular. Los datos personales no podrán ser obtenidos o divulgados sin previa autorización, o en ausencia de mandato legal o judicial que releve el consentimiento.
- d) **Principio de veracidad o calidad:** La información sujeta a Tratamiento debe ser veraz, completa, exacta, actualizada, comprobable y comprensible. Se prohíbe el Tratamiento de datos parciales, incompletos, fraccionados o que induzcan a error.
- e) **Principio de transparencia:** En el Tratamiento debe garantizarse el derecho del Titular a obtener del responsable del Tratamiento o del Encargado del Tratamiento, en cualquier momento y sin restricciones, información acerca de la existencia de datos que le conciernan.

- f) Principio de acceso y circulación restringida:** El Tratamiento se sujeta a los límites que se derivan de la naturaleza de los datos personales, de las disposiciones de la presente ley y la Constitución. En este sentido, el Tratamiento sólo podrá hacerse por personas autorizadas por el Titular y/o por las personas previstas en la presente ley.
- g) Datos personales:** Los datos personales, salvo la información pública, no podrán estar disponibles en Internet u otros medios de divulgación o comunicación masiva, salvo que el acceso sea técnicamente controlable para brindar un conocimiento restringido sólo a los Titulares o terceros autorizados conforme a la presente ley.
- h) Principio de seguridad:** La información sujeta a Tratamiento por el responsable del Tratamiento o Encargado del Tratamiento a que se refiere la presente ley, se deberá manejar con las medidas técnicas, humanas y administrativas que sean necesarias para otorgar seguridad a los registros evitando su adulteración, pérdida, consulta, uso o acceso no autorizado o fraudulento.
- i) Principio de confidencialidad:** Todas las personas que intervengan en el Tratamiento de datos personales que no tengan la naturaleza de públicos están obligadas a garantizar la reserva de la información, inclusive después de finalizada su relación con alguna de las labores que comprende el Tratamiento, pudiendo sólo realizar suministro o comunicación de datos personales cuando ello corresponda al desarrollo de las actividades autorizadas en la presente ley y en los términos de la misma.

3.4.4 Título III Categorías especiales de datos

3.4.4.1 Artículo 5°. Datos sensibles. Para los propósitos de la presente ley, se entiende por datos sensibles aquellos que afectan la intimidad del Titular o cuyo uso indebido puede generar su discriminación, tales como aquellos que revelen el origen racial o étnico, la orientación política, las convicciones religiosas o filosóficas, la pertenencia a sindicatos, organizaciones sociales, de derechos humanos o que promueva intereses de cualquier partido político o que garanticen los derechos y garantías de partidos políticos de oposición así como los datos relativos a la salud, a la vida sexual y los datos biométricos.

3.4.4.2 Artículo 6°. Tratamiento de datos sensibles. Se prohíbe el Tratamiento de datos sensibles, excepto cuando:

- a) El Titular haya dado su autorización explícita a dicho Tratamiento, salvo en los casos que por ley no sea requerido el otorgamiento de dicha autorización.
- b) El Tratamiento sea necesario para salvaguardar el interés vital del Titular y este se encuentre física o jurídicamente incapacitado. En estos eventos, los representantes legales deberán otorgar su autorización.
- c) El Tratamiento sea efectuado en el curso de las actividades legítimas y con las debidas garantías por parte de una fundación, ONG, asociación o cualquier otro organismo sin ánimo de lucro, cuya finalidad sea política, filosófica, religiosa o sindical, siempre que se refieran exclusivamente a sus miembros o a las personas que mantengan contactos regulares por razón de su finalidad. En estos eventos, los datos no se podrán suministrar a terceros sin la autorización del Titular.

- d) El Tratamiento se refiera a datos que sean necesarios para el reconocimiento, ejercicio o defensa de un derecho en un proceso judicial.
- e) El Tratamiento tenga una finalidad histórica, estadística o científica. En este evento deberán adoptarse las medidas conducentes a la supresión de identidad de los Titulares.

3.4.4.3 Artículo 7°. Derechos de los niños, niñas y adolescentes. En el Tratamiento se asegurará el respeto a los derechos prevalentes de los niños, niñas y adolescentes.

Queda proscrito el Tratamiento de datos personales de niños, niñas y adolescentes, salvo aquellos datos que sean de naturaleza pública.

Es tarea del Estado y las entidades educativas de todo tipo proveer información y capacitar a los representantes legales y tutores sobre los eventuales riesgos a los que se enfrentan los niños, niñas y adolescentes respecto del Tratamiento indebido de sus datos personales, y proveer de conocimiento acerca del uso responsable y seguro por parte de niños, niñas y adolescentes de sus datos personales, su derecho a la privacidad y protección de su información personal y la de los demás. El Gobierno Nacional reglamentará la materia, dentro de los seis (6) meses siguientes a la promulgación de esta ley.

3.4.5 Título IV Derechos y condiciones de legalidad para el tratamiento de datos

3.4.5.1 Artículo 8°. Derechos de los titulares. El Titular de los datos personales tendrá los siguientes derechos:

- a) Conocer, actualizar y rectificar sus datos personales frente a los responsables del Tratamiento o Encargados del Tratamiento. Este derecho se podrá ejercer, entre otros frente a datos parciales, inexactos, incompletos, fraccionados, que induzcan a error, o aquellos cuyo Tratamiento esté expresamente prohibido o no haya sido autorizado.
- b) Solicitar prueba de la autorización otorgada al responsable del Tratamiento salvo cuando expresamente se exceptúe como requisito para el Tratamiento, de conformidad con lo previsto en el artículo 10 de la presente ley.
- c) Ser informado por el responsable del Tratamiento o el Encargado del Tratamiento, previa solicitud, respecto del uso que les ha dado a sus datos personales.
- d) Presentar ante la Superintendencia de Industria y Comercio quejas por infracciones a lo dispuesto en la presente ley y las demás normas que la modifiquen, adicionen o complementen.
- e) Revocar la autorización y/o solicitar la supresión del dato cuando en el Tratamiento no se respeten los principios, derechos y garantías constitucionales y legales. La revocatoria y/o supresión procederá cuando la Superintendencia de Industria y Comercio haya determinado que en el Tratamiento el responsable o Encargado han incurrido en conductas contrarias a esta ley y a la Constitución.
- f) Acceder en forma gratuita a sus datos personales que hayan sido objeto de Tratamiento.

3.4.5.2 Artículo 9º. Autorización del titular. Sin perjuicio de las excepciones previstas en la ley, en el Tratamiento se requiere la autorización previa e informada del Titular, la cual deberá ser obtenida por cualquier medio que pueda ser objeto de consulta posterior.

3.4.5.3 Artículo 10. Casos en que no es necesaria la autorización. La autorización del Titular no será necesaria cuando se trate de:

- a) Información requerida por una entidad pública o administrativa en ejercicio de sus funciones legales o por orden judicial.
- b) Datos de naturaleza pública
- c) Casos de urgencia médica o sanitaria.
- d) Tratamiento de información autorizado por la ley para fines históricos, estadísticos o científicos.
- e) Datos relacionados con el Registro Civil de las Personas.

Quien acceda a los datos personales sin que medie autorización previa deberá en todo caso cumplir con las disposiciones contenidas en la presente ley.

3.4.5.4 Artículo 11. Suministro de la información. La información solicitada podrá ser suministrada por cualquier medio, incluyendo los electrónicos, según lo requiera el Titular. La información deberá ser de fácil lectura, sin barreras técnicas que impidan su acceso y deberá corresponder en un todo a aquella que repose en la base de datos.

El Gobierno Nacional establecerá la forma en la cual los responsables del Tratamiento y Encargados del Tratamiento deberán suministrar la información del Titular, atendiendo a la naturaleza del dato personal, Esta reglamentación deberá darse a más tardar dentro del año siguiente a la promulgación de la presente ley.

3.4.5.5 Artículo 12. Deber de informar al titular. El responsable del Tratamiento, al momento de solicitar al Titular la autorización, deberá informarle de manera clara y expresa lo siguiente:

- a) El Tratamiento al cual serán sometidos sus datos personales y la finalidad del mismo.
- b) El carácter facultativo de la respuesta a las preguntas que le sean hechas, cuando estas versen sobre datos sensibles o sobre los datos de las niñas, niños y adolescentes.
- c) Los derechos que le asisten como Titular.
- d) La identificación, dirección física o electrónica y teléfono del responsable del Tratamiento.

Parágrafo. El responsable del Tratamiento deberá conservar prueba del cumplimiento de lo previsto en el presente artículo y, cuando el Titular lo solicite, entregarle copia de esta.

3.4.5.6 Artículo 13. Personas a quienes se les puede suministrar la información. La información que reúna las condiciones establecidas en la presente ley podrá suministrarse a las siguientes personas:

- a) A los Titulares, sus causahabientes o sus representantes legales.

- b) A las entidades públicas o administrativas en ejercicio de sus funciones legales o por orden judicial.
- c) A los terceros autorizados por el Titular o por la ley.

3.4.6 Título V Procedimientos

3.4.6.1 Artículo 14. Consultas. Los Titulares o sus causahabientes podrán consultar la información personal del Titular que repose en cualquier base de datos, sea esta del sector público o privado. El responsable del Tratamiento o Encargado del Tratamiento deberán suministrar a estos toda la información contenida en el registro individual o que esté vinculada con la identificación del Titular.

La consulta se formulará por el medio habilitado por el responsable del Tratamiento o Encargado del Tratamiento, siempre y cuando se pueda mantener prueba de esta. La consulta será atendida en un término máximo de diez (10) días hábiles contados a partir de la fecha de recibo de la misma. Cuando no fuere posible atender la consulta dentro de dicho término, se informará al interesado, expresando los motivos de la demora y señalando la fecha en que se atenderá su consulta, la cual en ningún caso podrá superar los cinco (5) días hábiles siguientes al vencimiento del primer término.

Parágrafo. Las disposiciones contenidas en leyes especiales o los reglamentos expedidos por el Gobierno Nacional podrán establecer términos inferiores, atendiendo a la naturaleza del dato personal.

3.4.6.2 Artículo 15. Reclamos. El Titular o sus causahabientes que consideren que la información contenida en una base de datos debe ser objeto de corrección, actualización o supresión, o cuando adviertan el presunto incumplimiento de cualquiera de los deberes contenidos en esta ley, podrán presentar un reclamo ante el responsable del Tratamiento o el Encargado del Tratamiento el cual será tramitado bajo las siguientes reglas:

El reclamo se formulará mediante solicitud dirigida al responsable del Tratamiento o al Encargado del Tratamiento, con la identificación del Titular, la descripción de los hechos que dan lugar al reclamo, la dirección, y acompañando los documentos que se quiera hacer valer. Si el reclamo resulta incompleto, se requerirá al interesado dentro de los cinco (5) días siguientes a la recepción del reclamo para que subsane las fallas. Transcurridos dos (2) meses desde la fecha del requerimiento, sin que el solicitante presente la información requerida, se entenderá que ha desistido del reclamo.

En caso de que quien reciba el reclamo no sea competente para resolverlo, dará traslado a quien corresponda en un término máximo de dos (2) días hábiles e informará de la situación al interesado.

1. Una vez recibido el reclamo completo, se incluirá en la base de datos una leyenda que diga "reclamo en trámite" y el motivo del mismo, en un término no mayor a dos (2) días hábiles. Dicha leyenda deberá mantenerse hasta que el reclamo sea decidido.

2. El término máximo para atender el reclamo será de quince (15) días hábiles contados a partir del día siguiente a la fecha de su recibo. Cuando no fuere posible atender el reclamo dentro de dicho término, se informará al interesado los motivos de la demora y la fecha en que se atenderá su reclamo, la cual en ningún caso podrá superar los ocho (8) días hábiles siguientes al vencimiento del primer término.

3.4.6.3 Artículo 16. Requisito de procedibilidad. El Titular o causahabiente sólo podrá elevar queja ante la Superintendencia de Industria y Comercio una vez haya agotado el trámite de consulta o reclamo ante el responsable del Tratamiento o Encargado del Tratamiento.

3.4.7 Título VI Deberes de los responsables del tratamiento y encargados del tratamiento

3.4.7.1 Artículo 17. Deberes de los responsables del tratamiento. Los responsables del Tratamiento deberán cumplir los siguientes deberes, sin perjuicio de las demás disposiciones previstas en la presente ley y en otras que rijan su actividad:

- a) Garantizar al Titular, en todo tiempo, el pleno y efectivo ejercicio del derecho de hábeas data.
- b) Solicitar y conservar, en las condiciones previstas en la presente ley, copia de la respectiva autorización otorgada por el Titular.
- c) Informar debidamente al Titular sobre la finalidad de la recolección y los derechos que le asisten por virtud de la autorización otorgada.

- d) Conservar la información bajo las condiciones de seguridad necesarias para impedir su adulteración, pérdida, consulta, uso o acceso no autorizado o fraudulento.
- e) Garantizar que la información que se suministre al Encargado del Tratamiento sea veraz, completa, exacta, actualizada, comprobable y comprensible.
- f) Actualizar la información, comunicando de forma oportuna al Encargado del Tratamiento, todas las novedades respecto de los datos que previamente le haya suministrado y adoptar las demás medidas necesarias para que la información suministrada a este se mantenga actualizada.
- g) Rectificar la información cuando sea incorrecta y comunicar lo pertinente al Encargado del Tratamiento.
- h) Suministrar al Encargado del Tratamiento, según el caso, únicamente datos cuyo Tratamiento esté previamente autorizado de conformidad con lo previsto en la presente ley.
- i) Exigir al Encargado del Tratamiento en todo momento, el respeto a las condiciones de seguridad y privacidad de la información del Titular.
- j) Tramitar las consultas y reclamos formulados en los términos señalados en la presente ley; k) Adoptar un manual interno de políticas y procedimientos para garantizar el adecuado cumplimiento de la presente ley y en especial, para la atención de consultas y reclamos.
- k) Informar al Encargado del Tratamiento cuando determinada información se encuentra en discusión por parte del Titular, una vez se haya presentado la reclamación y no haya finalizado el trámite respectivo.
- l) Informar a solicitud del Titular sobre el uso dado a sus datos.

- m) Informar a la autoridad de protección de datos cuando se presenten violaciones a los códigos de seguridad y existan riesgos en la administración de la información de los Titulares.
- n) Cumplir las instrucciones y requerimientos que imparta la Superintendencia de Industria y Comercio.

3.4.7.2 Artículo 18. Deberes de los encargados del tratamiento. Los Encargados del Tratamiento deberán cumplir los siguientes deberes, sin perjuicio de las demás disposiciones previstas en la presente ley y en otras que rijan su actividad:

- a) Garantizar al Titular, en todo tiempo, el pleno y efectivo ejercicio del derecho de hábeas data.
- b) Conservar la información bajo las condiciones de seguridad necesarias para impedir su adulteración, pérdida, consulta, uso o acceso no autorizado o fraudulento.
- c) Realizar oportunamente la actualización, rectificación o supresión de los datos en los términos de la presente ley.
- d) Actualizar la información reportada por los responsables del Tratamiento dentro de los cinco (5) días hábiles contados a partir de su recibo.
- e) Tramitar las consultas y los reclamos formulados por los Titulares en los términos señalados en la presente ley.
- f) Adoptar un manual interno de políticas y procedimientos para garantizar el adecuado cumplimiento de la presente ley y, en especial, para la atención de consultas y reclamos por parte de los Titulares.

- g) Registrar en la base de datos las leyendas "reclamo en trámite" en la forma en que se regula en la presente ley.
- h) Insertar en la base de datos la leyenda "información en discusión judicial" una vez notificado por parte de la autoridad competente sobre procesos judiciales relacionados con la calidad del dato personal.
- i) Abstenerse de circular información que esté siendo controvertida por el Titular y cuyo bloqueo haya sido ordenado por la Superintendencia de Industria y Comercio.
- j) Permitir el acceso a la información únicamente a las personas que pueden tener acceso a ella.
- k) Informar a la Superintendencia de Industria y Comercio cuando se presenten violaciones a los códigos de seguridad y existan riesgos en la administración de la información de los Titulares.
- l) Cumplir las instrucciones y requerimientos que imparta la Superintendencia de Industria y Comercio.

Parágrafo. En el evento en que concurren las calidades de responsable del Tratamiento y Encargado del Tratamiento en la misma persona, le será exigible el cumplimiento de los deberes previstos para cada uno.

3.4.8 Título VII de los mecanismos de vigilancia y sanción. Capítulo I. De la autoridad de protección de datos

3.4.8.1 Artículo 19. Autoridad de protección de datos. La Superintendencia de Industria y Comercio, a través de una Delegatura para la Protección de Datos Personales, ejercerá la vigilancia para garantizar que en el Tratamiento de datos personales se respeten los principios, derechos, garantías y procedimientos previstos en la presente ley.

Parágrafo 1º. El Gobierno Nacional en el plazo de seis (6) meses contados a partir de la fecha de entrada en vigencia de la presente ley incorporará dentro de la estructura de la Superintendencia de Industria y Comercio un despacho de Superintendente Delegado para ejercer las funciones de Autoridad de Protección de Datos.

Parágrafo 2º. La vigilancia del tratamiento de los datos personales regulados en la Ley 1266 de 2008 se sujetará a lo previsto en dicha norma.

3.4.8.2 Artículo 20. Recursos para el ejercicio de sus funciones. La Superintendencia de Industria y Comercio contará con los siguientes recursos para ejercer las funciones que le son atribuidas por la presente ley:

- a) Los recursos que le sean destinados en el Presupuesto General de la Nación.

3.4.8.3 Artículo 21. Funciones. La Superintendencia de Industria y Comercio ejercerá las siguientes funciones:

- a) Velar por el cumplimiento de la legislación en materia de protección de datos personales.
- b) Adelantar las investigaciones del caso, de oficio o a petición de parte y, como resultado de ellas, ordenar las medidas que sean necesarias para hacer efectivo el derecho de hábeas data. Para el efecto, siempre que se desconozca el derecho, podrá disponer que se conceda el acceso y suministro de los datos, la rectificación, actualización o supresión de los mismos.
- c) Disponer el bloqueo temporal de los datos cuando, de la solicitud y de las pruebas aportadas por el Titular, se identifique un riesgo cierto de vulneración de sus derechos fundamentales, y dicho bloqueo sea necesario para protegerlos mientras se adopta una decisión definitiva.
- d) Promover y divulgar los derechos de las personas en relación con el Tratamiento de datos personales e implementara campañas pedagógicas para capacitar e informar a los ciudadanos acerca del ejercicio y garantía del derecho fundamental a la protección de datos.
- e) Impartir instrucciones sobre las medidas y procedimientos necesarios para la adecuación de las operaciones de los responsables del Tratamiento y Encargados del Tratamiento a las disposiciones previstas en la presente ley.
- f) Solicitar a los responsables del Tratamiento y Encargados del Tratamiento la información que sea necesaria para el ejercicio efectivo de sus funciones.
- g) Proferir las declaraciones de conformidad sobre las transferencias internacionales de datos.

- h) Administrar el Registro Nacional Público de Bases de Datos y emitir las órdenes y los actos necesarios para su administración y funcionamiento.
- i) Sugerir o recomendar los ajustes, correctivos o adecuaciones a la normatividad que resulten acordes con la evolución tecnológica, informática o comunicacional.
- j) Requerir la colaboración de entidades internacionales o extranjeras cuando se afecten los derechos de los Titulares fuera del territorio colombiano con ocasión, entre otras, de la recolección internacional de datos personales.
- k) Las demás que le sean asignadas por ley.

3.4.9 Capítulo II Procedimiento y Sanciones

3.4.9.1 Artículo 22. Trámite. La Superintendencia de Industria y Comercio, una vez establecido el incumplimiento de las disposiciones de la presente ley por parte del responsable del Tratamiento o el Encargado del Tratamiento, adoptará las medidas o impondrá las sanciones correspondientes.

En lo no reglado por la presente ley y los procedimientos correspondientes se seguirán las normas pertinentes del Código Contencioso Administrativo.

3.4.9.2 Artículo 23. Sanciones. La Superintendencia de Industria y Comercio podrá imponer a los responsables del Tratamiento y Encargados del Tratamiento las siguientes sanciones:

- a) Multas de carácter personal e institucional hasta por el equivalente de dos mil (2.000) salarios mínimos mensuales legales vigentes al momento de la imposición de la sanción. Las multas podrán ser sucesivas mientras subsista el incumplimiento que las originó.
- b) Suspensión de las actividades relacionadas con el Tratamiento hasta por un término de seis (6) meses. En el acto de suspensión se indicarán los correctivos que se deberán adoptar.
- c) Cierre temporal de las operaciones relacionadas con el Tratamiento una vez transcurrido el término de suspensión sin que se hubieren adoptado los correctivos ordenados por la Superintendencia de Industria y Comercio.
- d) Cierre inmediato y definitivo de la operación que involucre el Tratamiento de datos sensibles.

Parágrafo. Las sanciones indicadas en el presente artículo sólo aplican para las personas de naturaleza privada. En el evento en el cual la Superintendencia de Industria y Comercio advierta un presunto incumplimiento de una autoridad pública a las disposiciones de la presente ley, remitirá la actuación a la Procuraduría General de la Nación para que adelante la investigación respectiva.

3.4.9.3 Artículo 24. Criterios para graduar las sanciones. Las sanciones por infracciones a las que se refiere el artículo anterior, se graduarán atendiendo los siguientes criterios, en cuanto resulten aplicables:

- a) La dimensión del daño o peligro a los intereses jurídicos tutelados por la presente ley.
- b) El beneficio económico obtenido por el infractor o terceros, en virtud de la comisión de la infracción.
- c) La reincidencia en la comisión de la infracción.
- d) La resistencia, negativa u obstrucción a la acción investigadora o de vigilancia de la Superintendencia de Industria y Comercio.
- e) La renuencia o desacato a cumplir las órdenes impartidas por la Superintendencia de Industria y Comercio.
- f) El reconocimiento o aceptación expresas que haga el investigado sobre la comisión de la infracción antes de la imposición de la sanción a que hubiere lugar.

3.4.10 Capítulo III del Registro nacional de bases de datos

3.4.10.1 Artículo 25. Definición. Reglamentado por el Decreto Nacional 886 de 2014 El Registro Nacional de Bases de Datos es el directorio público de las bases de datos sujetas a Tratamiento que operan en el país.

El registro será administrado por la Superintendencia de Industria y Comercio y será de libre consulta para los ciudadanos.

Para realizar el registro de bases de datos, los interesados deberán aportar a la Superintendencia de Industria y Comercio las políticas de tratamiento de la información, las cuales obligarán a los responsables y encargados del mismo, y cuyo incumplimiento acarreará las

sanciones correspondientes. Las políticas de Tratamiento en ningún caso podrán ser inferiores a los deberes contenidos en la presente ley.

Parágrafo. El Gobierno Nacional reglamentará, dentro del año siguiente a la promulgación de la presente ley, la información mínima que debe contener el Registro, y los términos y condiciones bajo los cuales se deben inscribir en este los responsables del Tratamiento.

3.4.11 Título VIII Transferencia de datos a terceros países

3.4.11.1 Artículo 26. Prohibición. Se prohíbe la transferencia de datos personales de cualquier tipo a países que no proporcionen niveles adecuados de protección de datos. Se entiende que un país ofrece un nivel adecuado de protección de datos cuando cumpla con los estándares fijados por la Superintendencia de Industria y Comercio sobre la materia, los cuales en ningún caso podrán ser inferiores a los que la presente ley exige a sus destinatarios.

Esta prohibición no regirá cuando se trate de:

1. Información respecto de la cual el Titular haya otorgado su autorización expresa e inequívoca para la transferencia.
2. Intercambio de datos de carácter médico, cuando así lo exija el Tratamiento del Titular por razones de salud o higiene pública.
3. Transferencias bancarias o bursátiles, conforme a la legislación que les resulte aplicable.

4. Transferencias acordadas en el marco de tratados internacionales en los cuales la República de Colombia sea parte, con fundamento en el principio de reciprocidad.
5. Transferencias necesarias para la ejecución de un contrato entre el Titular y el responsable del Tratamiento, o para la ejecución de medidas precontractuales siempre y cuando se cuente con la autorización del Titular.
6. Transferencias legalmente exigidas para la salvaguardia del interés público, o para el reconocimiento, ejercicio o defensa de un derecho en un proceso judicial.

Parágrafo 1º. En los casos no contemplados como excepción en el presente artículo, corresponderá a la Superintendencia de Industria y Comercio, proferir la declaración de conformidad relativa a la transferencia internacional de datos personales. Para el efecto, el Superintendente queda facultado para requerir información y adelantar las diligencias tendientes a establecer el cumplimiento de los presupuestos que requiere la viabilidad de la operación.

Parágrafo 2º. Las disposiciones contenidas en el presente artículo serán aplicables para todos los datos personales, incluyendo aquellos contemplados en la Ley 1266 de 2008.

3.4.12 Título IX Otras disposiciones

3.4.12.1 Artículo 27. Normas corporativas vinculante. El Gobierno Nacional expedirá la reglamentación correspondiente sobre Normas Corporativas Vinculantes para la certificación de buenas prácticas en protección de datos, personales y su transferencia a terceros países.

3.4.12.2 Artículo 28. Régimen de transición. Las personas que a la fecha de entrada en vigencia de la presente ley ejerzan alguna de las actividades acá reguladas tendrán un plazo de hasta seis (6) meses para adecuarse a las disposiciones contempladas en esta ley.

3.4.12.3 Artículo 29. Derogatorias. La presente ley deroga todas las disposiciones que le sean contrarias a excepción de aquellas contempladas en el artículo 2°. **Artículo 30. Vigencia.** La presente ley rige a partir de su promulgación.

Capítulo 4. Diseño metodológico

4.1 Tipo de investigación

“El enfoque cualitativo se selecciona cuando el propósito es examinar la forma en que los individuos perciben y experimentan los fenómenos que los rodean, profundizando en sus puntos de vista, interpretaciones y significados”. (Hernández, 2014)

Con base en este concepto, para el desarrollo de la investigación se hará uso de la metodología cualitativa descriptiva, a fin de proporcionar soporte investigativo al modelo de desarrollo de prototipos, buscando la caracterización de las experiencias de usuario en el sistema de control de seguridad del Edificio de Ingeniería y la creación de un entorno de desarrollo eficiente y eficaz.

4.2 Población y muestra

4.2.1 Población

La población objeto de estudio de la investigación está conformada por los diferentes estamentos universitarios que cuenten con acceso al Edificio de Ingeniería de la Universidad Francisco de Paula Santander Ocaña, como los estudiantes y administrativos del recinto.

4.2.2 Muestra

Al iniciar la investigación se tomará una muestra selectiva de la población con base en el contexto de participación de los individuos que forman parte del sistema, variando la muestra según sea necesario.

4.3 Estrategia metodológica

4.3.1 Modelo de desarrollo de prototipos

“El Modelo de Prototipos, llamado también Modelo de Desarrollo Evolutivo facilita al constructor del software y al usuario final tener una perspectiva a futuro del producto a desarrollar, ya que se tiene una vista previa del software en términos generales”. (EcuRed, EcuRed, 2013)

“Un prototipo es una versión inicial de un sistema de software que se usa para demostrar conceptos, tratar opciones de diseño y encontrar más sobre el problema y sus posibles soluciones”. (CGN, 2017)

La idea principal es ir moldeando este prototipo durante un proceso de revisión de N versiones, cumpliendo con los objetivos específicos planteados hasta que se llegue al producto deseado, de ser necesario se tendrá que repetir el proceso tal como en el modelo de espiral. Esto mediante proceso iterativo donde en cada ciclo se tendrá un punto de partida

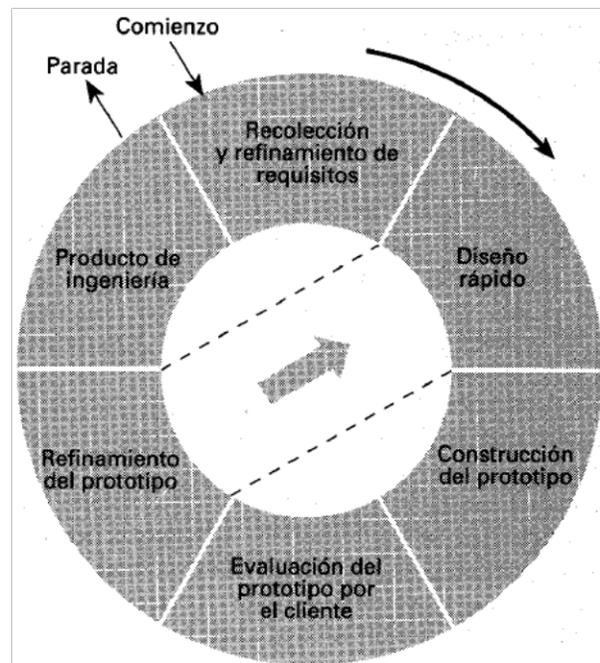
(requerimientos) del cual surgirá una pre visualización del producto deseado (prototipo rápido), seguido de la evolución del proyecto (desarrollo) que permita al final del ciclo obtener el producto de ingeniería deseado. (Alonso, Martínez, & Segovia, 2005)

“En el proceso de ingeniería de requerimientos, un prototipo ayuda con la selección y validación de requerimientos del sistema”. (CGN, 2017)

“En el proceso de diseño de sistemas, un prototipo sirve para buscar soluciones específicas de software y apoyar el diseño de interfaces del usuario”. (EcuRed, 2021)

Figura 2 Modelo de prototipos.

Modelo de prototipos.



Nota. El gráfico muestra las fases de aplicación del modelo de desarrollo de prototipos. Datos tomados de (EcuRed, 2021)

4.3.1.1 Recolección y refinamiento de requisitos. Debido a la cualidad investigativa del proyecto, para esta etapa se tiene predestinado el planteamiento de requerimientos teniendo en cuenta el aporte de los investigadores, en función de la solución a la problemática. Para lograr dicha finalidad, los investigadores realizarán un análisis del entorno y determinarán las necesidades del mismo, obteniendo un listado de requerimientos base para la determinación de herramientas y la construcción del prototipo en su fase inicial.

4.3.1.2 Diseño rápido. Una vez realizada la recolección de requerimientos se planteará un diseño del prototipo que supla las necesidades funcionales del mismo, pero que no se enfoque en un resultado altamente estético, con la finalidad de llevar a cabo un desarrollo ágil en comparación con un desarrollo mayormente estructurado dentro de otra metodología, brindando a los investigadores la retroalimentación necesaria para realizar modificaciones a largo plazo del prototipo funcional, según se requiera.

4.3.1.3 Construcción del prototipo. Ya adquiridos los requerimientos y estipulado el diseño ágil, independientemente de la etapa evolutiva del proyecto investigativo, se procederá con la construcción del prototipo, codificando según la necesidad de los requerimientos y guiados por la etapa de diseño ágil, facilitando así todo el proceso de construcción, permitiendo una beta evolutiva del software y el proyecto en general.

4.3.1.4 Evaluación del prototipo por el cliente. Debido al enfoque y la proyección investigativa, para esta etapa según la determinación de los requerimientos por parte de los

investigadores, se realizará una evaluación efectiva respecto al entorno de estudio, teniendo como referente que la metodología de desarrollo de prototipos brinda las herramientas necesarias para que se dé este ambiente maleable en cuanto a todas las etapas y criterios de ejecución del proyecto.

4.3.1.5 Refinamiento del prototipo. Es así como en esta etapa de la metodología, se realizará una retroalimentación de los resultados obtenidos en las etapas anteriores y se realizará un análisis con la finalidad de promover nuevas estructuras que permitan el mejoramiento del prototipo resultante. Si se determina que se requiere realizar mejoras al prototipo se retorna a la etapa de diseño rápido con la finalidad de iterar de nuevo sobre las demás etapas subsecuentes.

4.3.1.6 Producto de ingeniería. Una vez determinado que el prototipo satisface las necesidades y requerimientos planteados desde un principio, se procede a realizar la entrega del prototipo final.

4.4 Técnica e instrumentos de recolección

Para el desarrollo de la investigación se hará uso de la observación directa como instrumento de recolección de datos, apoyados en el comportamiento e interacción con el prototipo planteado según su etapa de desarrollo, con la finalidad de realizar una retroalimentación para su mejoramiento en la siguiente sucesión de etapas.

A su vez, si se requiere o no, se realizará el uso de la entrevista a las personas que interactúen con el prototipo como apoyo a los procesos de desarrollo.

4.5 Análisis de la Información

El análisis se realizará por medio de la digitación de la información recolectada, donde se determinarán diversos puntos focales que nos permitirán precisión en la estructuración de los requerimientos funcionales y no funcionales para el desarrollo del prototipo.

Capítulo 5. Resultados

5.1 Estado del arte

5.1.1 *Biometría*

“La palabra biometría deriva del griego bios (que quiere decir **vida**) y metría (que quiere decir **medida**). Los rasgos biométricos son medidas extraídas del cuerpo humano vivo. Y, además, todos los rasgos biométricos son una combinación de anatomía y de comportamiento”. (Serratosa, 2008)

“La biometría es una ciencia que analiza las distancias y posiciones entre las partes del cuerpo para poder identificar o clasificar personas” (Serratosa, 2008), también podría definirse como “la medición y análisis estadístico de las características físicas y de comportamiento único de las personas”. (Portilla Jiménez, 2018)

Existe una gran variedad de rasgos biométricos que han sido estudiados para la identificación de las personas a lo largo de la historia, uno de los que más ha tenido hincapié ha sido el método de reconocimiento por huella dactilar, de igual forma, existen muchos otros rasgos biométricos que han sido utilizados para el reconocimiento de las personas como: el rostro, el iris, las manos, el ADN, la retina y la firma. En la actualidad la implementación de la biometría en sistemas de seguridad ha fortalecido el control de acceso en zonas restringidas, garantizando la autenticidad de las personas que ingresan a los recintos.

Teniendo en cuenta la variedad de rasgos biométricos que existen para la identificación de las personas, se pueden implementar sistemas de seguridad altamente complejos que estudien en paralelo múltiples rasgos biométricos y permitan fortalecer su invulnerabilidad, evitando la identificación de falsos positivos dentro del sistema.

5.1.2 Historia de la Biometría

“En 1882, el policía francés Alphonse Bertillon (1853 - 1914) presentó el primer sistema de identificación de las personas basado en las características físicas, es decir, en los rasgos biométricos, y lo denominó antropometría.” (Serratosa, 2008) De esta manera surgió uno de los primeros sistemas científicos para el estudio de criminales, donde se clasificaban los rasgos del rostro y cuerpo de las personas. En sus inicios la biometría no se consideraba una ciencia importante para su estudio, pero luego de obtener grandes resultados en la identificación de criminales reincidentes, se apoyaron las investigaciones.

El sistema planteado por Bertillon consistía en guardar la información de once medidas físicas de las personas basadas en su cabeza y cuerpo, como son: (Serratosa, 2008)

1. Altura de la persona.
2. Anchura de los brazos extendidos.
3. Altura de la persona sentada.
4. Largura de la cabeza.

5. Anchura de la cabeza.
6. Largura de la oreja derecha.
7. Anchura de la oreja derecha.
8. Largura del pie izquierdo.
9. Largura del dedo corazón izquierdo.
10. Largura del dedo meñique de la mano izquierda.
11. Largura del antebrazo izquierdo.

De esta forma se podía calcular la similitud entre dos personas, determinando la distancia euclídea entre los vectores formados por los once componentes planteados en el sistema de Bertillon. Si se desea conocer si dos vectores pertenecen a la misma persona de acuerdo con las once medidas planteadas, siendo A y B dos de estos, se tiene que:

$$D_{Bertillon}(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^{11} (A_i - B_i)^2}$$

Donde se considera que pertenece a la misma persona si:

$$D_{Bertillon}(\mathbf{A}, \mathbf{B}) \leq L_{lindar}_{Bertillon}$$

A este estudio comparativo se le llamó bertillonaje en honor a Bertillon, en la siguiente figura se puede observar cómo se realizaba la captura y registro de la información de los posibles criminales:

Figura 3 Dibujos de Bertillon representativos de la toma de medidas biométricas.

Dibujos de Bertillon representativos de la toma de medidas biométricas.



Nota. La imagen muestra las once medidas que formaban la ficha de los infractores. Datos tomados de (Serratosa, 2008).

A pesar de que Bertillon estableció un método práctico para la identificación de personas de acuerdo con sus rasgos biométricos, en el momento de su aplicabilidad se dieron cuenta que con el pasar del tiempo, los rasgos biométricos que caracterizaban a las personas iban presentando alteraciones que dejaban de garantizar su autenticidad, por lo que, prefirieron usar

como método de identificación el estudio de las huellas dactilares ya que tenían bases científicas más robustas y no presentaban alteraciones.

Esta premisa se vio reflejada en el primer caso de inviabilidad del método de Bertillon, el caso de Will West y William West en el año de 1903, William West fue un criminal que fue arrestado y condenado a prisión en 1901, durante el proceso de encarcelamiento se le realizó el registro de las medidas de Bertillon, dejando un historial biométrico en el archivo policial. Unos años más tarde, en 1903, Will West contó con la misma suerte y fue arrestado por la policía, por lo que se le realizó el mismo proceso de registro de las medidas de Bertillon, al revisar y analizar el registro biométrico, dedujeron que Will West y William West eran la misma persona, pero había cambiado su nombre para ocultar su identidad, sin embargo no se percataron que William West ya se encontraba en prisión, por lo que el método de Bertillon fue duramente desprestigiado por este hecho.

Con el pasar de los años se presentaron casos de criminología que fueron resueltos con ayuda del análisis biométrico, en especial por el descubrimiento de huellas dactilares sanguinolentas en las escenas del crimen que evidenciaban el hecho cometido por el autor, siendo así como el Ministerio del Interior del Reino Unido aceptó que indiscutiblemente la huella dactilar era única de la persona y no podía verse duplicada en ninguna otra. De esta forma, se adoptó la huella dactilar como método imprescindible en las comisarías policiales, pudiendo comparar las huellas registradas con los archivos policiales para determinar si un criminal había vuelto a reincidir en un crimen que aseverara su sentencia.

A pesar de que el análisis de huellas dactilares fue práctico, llegó a un punto donde se volvió tedioso identificar a las personas registradas debido al exceso de información que se iba almacenando en el archivo policial, además, el proceso de búsqueda de las personas se realizaba de forma manual, por lo que el análisis dactilar pasó a ser una tarea demandante, debido a esto, se estableció una clasificación en el método de huellas dactilares que permitiera optimizar el proceso de búsqueda.

Aunque clasificar las huellas dactilares mejoró los tiempos de respuesta en la búsqueda, el proceso de identificación de las personas seguía siendo lento, ya que el análisis y comparación de la información debía ser meticulosa según la clase y tipo de huella dactilar, por ese motivo, se proyectó la aplicabilidad del desarrollo tecnológico dentro de esta área, estableciendo la automatización de los procesos analíticos que dieran como resultado sistemas automáticos de identificación de huellas dactilares con respuestas inmediatas de identificación.

5.1.3 Reconocimiento Biométrico

Desde siempre, la identificación ha sido necesaria para el reconocimiento personal por medio de simples tarjetas de identificación o códigos secretos, pero a pesar de estas formas de identificación su vulnerabilidad se ve fácilmente afectada por la suplantación, por lo que no existe una garantía de la identidad de las personas, por este motivo, la biometría se ha convertido en un factor crucial para la identificación, gracias a que los rasgos biométricos son únicos e inalterables y permiten una representación corporal de las personas, para posteriormente realizar una asociación de los rasgos biométricos con la identidad personal.

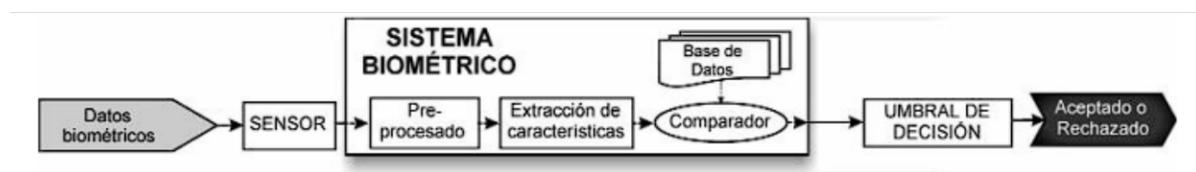
De esta forma, Serratos se refiere al reconocimiento biométrico como: “El uso de diferentes características anatómicas (huellas dactilares, cara o iris) y comportamiento (como habla, firma o teclear). Estas características se denominan identificadores biométricos o rasgos biométricos y sirven para reconocer automáticamente a los individuos.” (Serratos, 2008)

5.1.4 Sistemas de Reconocimiento Biométrico

“Un sistema biométrico esencialmente es un reconocedor de patrones que captura datos biométricos de un individuo, extrae un conjunto de características a partir de dichos datos y las compara con otros patrones previamente almacenados en el sistema”. (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008) Los sistemas de reconocimiento biométrico tienen una estructura con diferentes fases que dependen del tipo de patrón de reconocimiento.

Figura 4 *Arquitectura de un sistema de reconocimiento biométrico*

Arquitectura de un sistema de reconocimiento biométrico



Nota. Datos tomados de (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008).

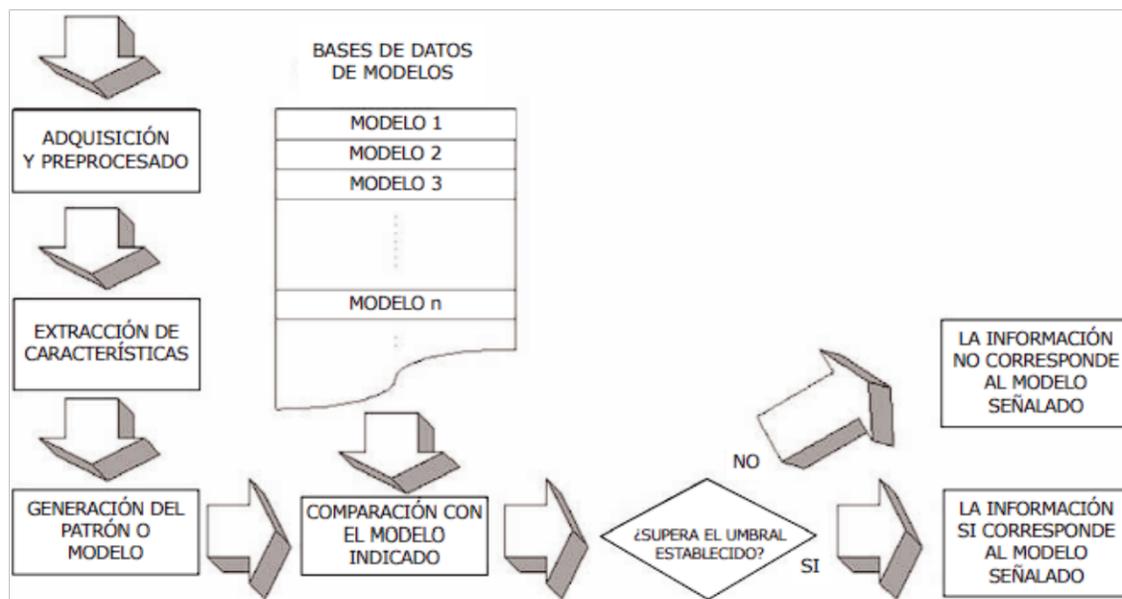
Existen dos tipos de sistemas de reconocimiento biométrico, el sistema de verificación y el sistema de identificación. El **sistema de verificación** se encarga de constatar que los rasgos

biométricos de la persona recién capturada por el sistema, correspondan con los rasgos biométricos de la persona registrada previamente dentro de los procesos de inscripción, autenticando su identificación.

Por lo general, los sistemas de verificación se caracterizan por tener resultados binarios, es decir: si los rasgos biométricos de la persona corresponden, se confirma que la persona ha sido identificada, de lo contrario, se determina que es una persona diferente. Para que el sistema funcione, el usuario debe presentar su identificación con sus rasgos biométricos y el sistema debe encargarse de validar que correspondan sus rasgos biométricos con la información biométrica almacenada en la base de datos.

Figura 5 Operación en modo verificación

Operación en modo verificación



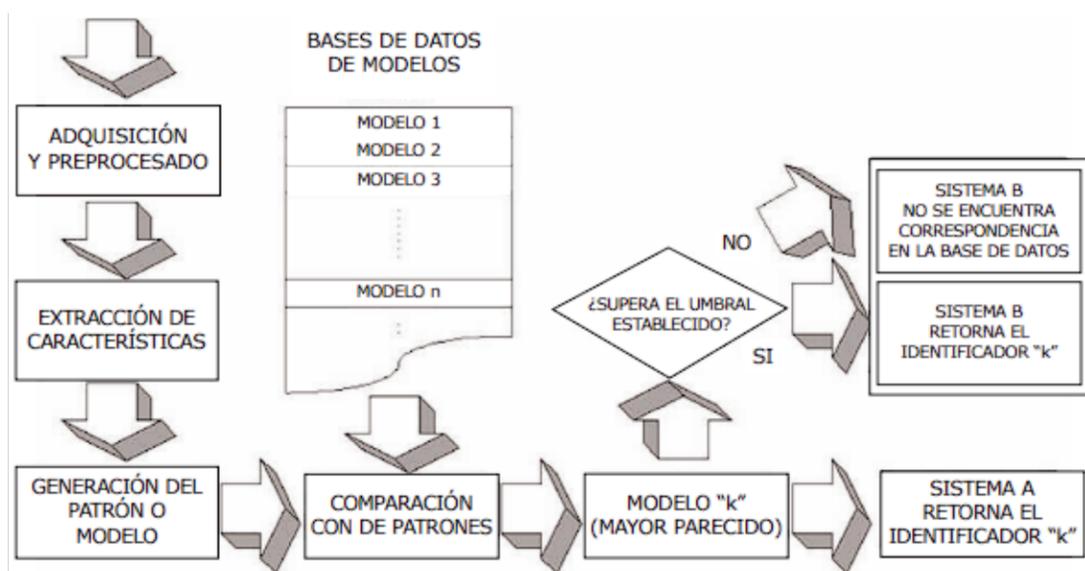
Nota. Datos tomados de (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008)

De igual forma, el **sistema de identificación** se encarga de verificar que los rasgos biométricos de las personas correspondan con la información almacenada en la base de datos, pero con la característica de que lo busca de acuerdo con un patrón de semejanza. En este sistema, los usuarios no aportan ningún tipo de identificación, el sistema por sí solo se encarga de realizar múltiples comparaciones biométricas en la base de datos que lo llevan a determinar la identidad de la persona.

El sistema de identificación, a diferencia del sistema de verificación, presenta tres tipos de posibles resultados, el primer resultado es la autenticación de la identidad de la persona, donde el rasgo biométrico coincide de acuerdo con el nivel de semejanza de la persona registrada según el umbral de posibilidades; en el segundo resultado, el sistema deduce que no existe ninguna semejanza de rasgo biométrico con los registrados dentro de la base de datos, debido a que se supera el umbral de posibilidades. A diferencia de los dos resultados anteriores, en el tercer resultado, el sistema no solo retorna una persona, sino que genera un listado de personas que poseen características similares, debido a que los rasgos biométricos pertenecen al umbral de posibilidades.

Figura 6 Operación en modo identificación

Operación en modo identificación



Nota. Datos tomados de (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008).

Para que el sistema de verificación tanto como el de identificación sean posibles, necesitan de un sistema en común: el **sistema de matriculación**. El sistema de matriculación es fundamental, debido a que se encarga del proceso de recolección y asociación de los rasgos biométricos de las personas con su identificación. Es importante resaltar que los procesos de recolección y asociación son altamente importantes, debido a qué al relacionar los rasgos biométricos con la identidad de la persona, no pueden existir inconsistencias; durante la ejecución de estos procesos, la persona encargada de registrar a los usuarios dentro del sistema debe velar por la integridad de la información, validando la representación de los datos suministrados.

Durante los procesos del sistema de matriculación, la persona encargada de registrar los datos biométricos tiene la potestad de determinar si la calidad de los rasgos biométricos cumplen con la necesidad del sistema, en caso de no cumplir, se realiza de nuevo el registro de los rasgos

biométricos hasta satisfacer la calidad requerida, para lograr esto, es necesario que se realice una serie de capturas del mismo rasgo biométrico y objetivamente se determine cual posee mejor calidad o se realice una fusión de todas las capturas efectuadas, que permita reducir el margen de error y no se presenten falsos positivos.

Tanto el sistema de matriculación, el sistema de verificación y el sistema de identificación, necesitan de varios procesos para ser efectuados:

- **Captura o Adquisición de datos:** En el proceso de captura se realiza una digitalización del rasgo biométrico, utilizando sensores que crean una imagen del rasgo biométrico que se requiere registrar, en muchas ocasiones, de acuerdo con el tipo de biometría, se usan otro tipo de sensores, además en muchos sistemas no solo se registra el rasgo biométrico, sino que se utilizan también otro tipo de artefactos complementarios que anexan información no biométrica dentro del sistema.
- **Preprocesamiento o Extracción de Características:** En ocasiones es necesario realizar un proceso de acondicionamiento que permita reducir las distorsiones que se generen en la información capturada o para normalizar la información en rangos específicos que brinden mayor efectividad en el proceso de reconocimiento. (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008). En esta etapa, se realiza un proceso de limpieza de la muestra biométrica debido a que la imagen digitalizada suele presentar altos porcentajes de ruido, para esto se aplican técnicas de compresión que permiten reducir la información en un conjunto de características discriminantes, únicas e

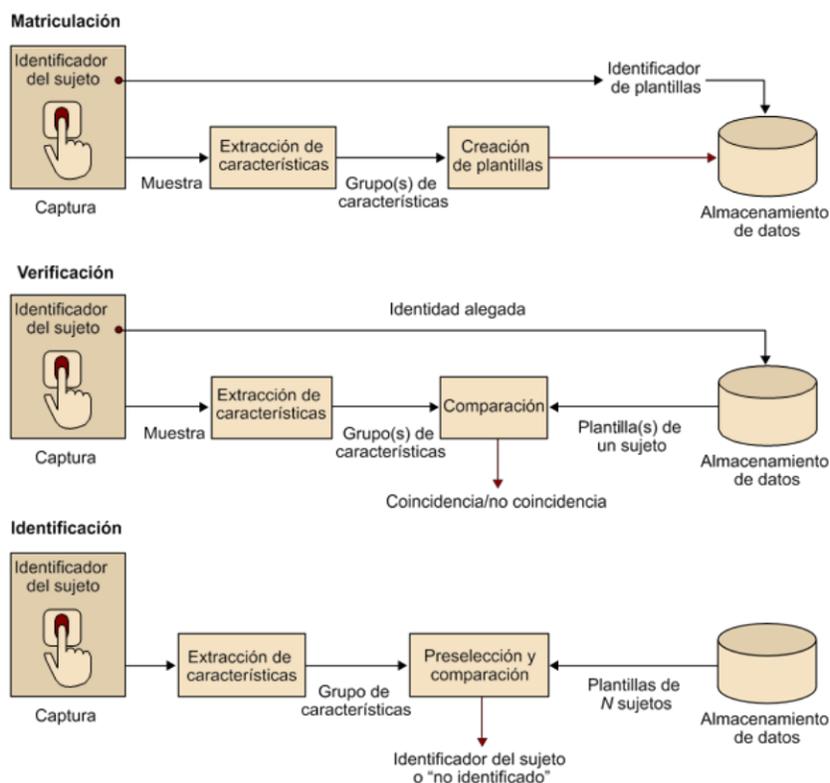
invariables de cada persona, garantizando la optimización de respuesta en el proceso de reconocimiento.

- **Creación de Plantilla o Creación de Modelos:** En los sistemas biométricos, la plantilla es considerada como una forma de representación del conjunto de muestras biométricas, como lo ejemplifica Serratos: “se puede crear una plantilla de dieciséis muestras diferentes de la imagen de la cara de una misma persona”. (Serratos, 2008) Para crear la plantilla se utiliza la información más característica del resultado de extracción, recopilando todos los registros de identificación y comprimiendo toda la información de los rasgos más característicos, según la plantilla que se esté realizando, puede llevar un nivel de complejidad más alto, en algunas ocasiones se pueden presentar plantillas con múltiples muestras o con una única muestra biométrica como registro de identificación.
- **Comparación o Umbral de decisión:** “El proceso de comparación recibe como entrada un registro de identificación y una plantilla y calcula una distancia entre los dos”. (Serratos, 2008) En ocasiones se realiza un proceso probabilístico que respalda la representatividad de una persona, es por esto que, en el proceso de verificación, existe un umbral dentro del sistema que determina el asertividad de las personas registradas. Como lo menciona Serratos, “si la distancia es inferior al umbral (o la probabilidad es superior al umbral), el sistema considera que los dos datos provienen de la misma persona, de lo contrario, se considera que provienen de dos personas diferentes.” (Serratos, 2008)
- **Filtración:** Debido al volumen de información de los sistemas biométricos, es necesario aplicar un método de filtración que permita reducir el tiempo de respuesta en la búsqueda de los registros del sistema, ya que los métodos tradicionales son menos eficientes.

- Almacenamiento de Datos:** Durante el proceso de almacenamiento es necesario registrar las características e información de los usuarios, por medio de un identificador único, una plantilla biométrica y un conjunto de información personal. En muchas aplicaciones, el proceso de almacenamiento se realiza de manera centralizada para efectuar el proceso de identificación o a través de tarjetas inteligentes para efectuar el proceso de verificación. Igualmente, se aplican técnicas de encriptación en el conjunto de datos de la información almacenada, para que el registro biométrico y la información personal formen un conjunto indivisible que garantice la seguridad de la relación de ambas partes.

Figura 7

Etapas o procesos que componen los sistemas: a. matriculación (enrollment); b. verificación (verification); c. identificación (identification)



Nota. Datos tomados de (Serratosa, 2008).

Los sistemas biométricos funcionan de dos formas, en línea y fuera de línea.

Los **sistemas biométricos en línea** son usualmente sistemas de verificación, por lo que necesitan ser eficientes y efectuar el proceso de comparación en tiempo real, debido a que necesitan respuestas inmediatas. Este tipo de sistemas se pueden apreciar en aplicaciones de validación de permisos o en aplicaciones de restricción de acceso a zonas no autorizadas.

Los **sistemas biométricos fuera de línea**, por el contrario, son usualmente sistemas de identificación y no requieren de respuestas en tiempo real, por lo que permite un margen de tolerancia al retraso en la respuesta. Este tipo de sistemas se pueden apreciar en aplicaciones de automatización, los cuales requieren que los rasgos biométricos sean capturados por medio de un sensor, de acuerdo al tipo de rasgo y sin el control de verificación de calidad de los datos. Normalmente este tipo de sistemas son semiautomáticos, es decir, el rasgo biométrico puede capturarse mediante un sistema electrónico o no y el control de calidad se realiza por medio de un especialista con ayuda de herramientas tecnológicas de comparación biométrica.

Así mismo, según el tipo de aplicación, se realizan dos tipos de búsquedas en los sistemas de identificación, las positivas y las negativas.

Las **búsquedas positivas** se caracterizan por comprobar que el rasgo biométrico capturado se encuentra almacenado en la base de datos, en otras palabras, si el usuario fue

matriculado con anterioridad, para conocer la identificación asociada al rasgo biométrico. Normalmente los datos se ingresan al sistema por medio de muestras dactilares o fotografías capturadas.

Por el contrario, las **búsquedas negativas**, están enfocadas en corroborar que la persona capturada no se encuentre matriculada en el sistema, es decir, que no exista ninguna asociación de los rasgos biométricos capturados con la información almacenada. Este tipo de sistemas son utilizados para comprobar que una persona no utilice más de una vez un servicio que es de único uso, por ejemplo, en un proceso electoral.

5.1.5 Rasgos Biométricos

Como se mencionó anteriormente, los rasgos biométricos son las características particulares que identifican a cualquier persona, por ende, existen dos tipos: la biometría estática y la biometría dinámica. La primera hace referencia a los rasgos biométricos que pueden ser extraídos del rostro, las orejas, la voz, el iris, el ADN, las huellas dactilares, las palmas de la mano y demás, en cambio, la segunda hace referencia a los rasgos biométricos conductuales, es decir, a todo tipo de gesto o acción que genera una persona en determinada situación, por ejemplo, la gesticulación, el habla, la forma de caminar, la escritura y la interacción con distintas herramientas. (Portilla Jiménez, 2018)

- Según (Serratosa, 2008) y (Portilla Jiménez, 2018). Los rasgos biométricos deben satisfacer los siguientes requerimientos para poder efectuar la verificación o la identificación de la persona:
 - **Universalidad:** Cada persona debe contar con el rasgo biométrico solicitado.
 - **Particularidad:** Todas las personas deben contar con un mínimo de diferencia en términos del rasgo biométrico.
 - **Permanencia:** El rasgo biométrico debe ser invariante en el tiempo y a cualquier otro factor, desde la comparativa de rasgos biométricos.
 - **Medible:** Los rasgos biométricos deben tener la característica de ser cuantificables, ya que necesitan ser codificados, registrados y procesados.
 - **Rendimiento:** De acuerdo con el nivel de requerimiento, el rasgo biométrico debe garantizar la precisión y robustez en diferentes entornos.
 - **Aceptabilidad:** Los usuarios del sistema deben aceptar el tratamiento de datos de su rasgo biométrico para su identificación.
 - **Infalsificable:** El rasgo biométrico debe garantizar que la usurpación de identidad sea dificultosa, para evitar situaciones fraudulentas como el robo de identidad o la modificación de rasgos biométricos.

5.1.6 Rasgos Biométricos Mayormente Utilizados

5.1.6.1 Rasgos biométricos de la cabeza.

5.1.6.1.1 Cara. La cara es uno de los rasgos biométricos mayormente aceptados debido a que es el rasgo de mayor naturalidad utilizado por los humanos para reconocer a una persona. Además, es un mecanismo no intrusivo para la adquisición de imágenes, por lo que no requiere una interacción directa con el sensor por parte del usuario. Actualmente los sistemas de reconocimiento de cara tienen un rendimiento aceptable en el uso comercial, sin embargo, se ven afectados por aspectos de luminosidad, entorno y postura de la cara, que de no ser controlados podrían reducir el rendimiento del sistema. (Serratosa, 2008) (Portilla Jiménez, 2018)

Algunos prototipos de reconocimiento no solo detectan a la persona, también detectan el estado de ánimo de acuerdo con su expresión facial, edad, sexo y posición; en estos sistemas, la captura de la imagen de la cara se lleva a cabo, a través de cámaras fotográficas o de vídeo. Por otra parte, existen sistemas que permiten reproducir una imagen en tres dimensiones con ayuda de un conjunto de múltiples cámaras posicionadas en diferentes ángulos y otras que contrarrestan las afectaciones de luminosidad con el uso de cámaras infrarrojas.

Figura 8 Reconocimiento de la cara

Reconocimiento de la cara

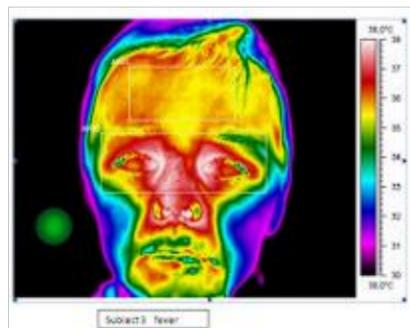


Nota. Datos tomados de (Serratosa, 2008).

5.1.6.1.2 Termograma facial. Con ayuda de un termógrafo junto con el reconocimiento de cara se puede llevar un control pasivo en casos muy específicos donde el usuario no sabe que está siendo identificado. Este tipo de reconocimiento no se ve afectado por maquillajes o cortes de cabello, pero tiene algunos inconvenientes como la afectación por resfriados o cambios de temperatura de la persona al realizar alguna actividad física. El rasgo biométrico se captura por medio de una cámara térmica situada a pocos metros de la persona.

Figura 9 Termograma facial

Termograma facial



Nota. Datos tomados de (Serratosa, 2008).

5.1.6.1.3 Oreja. La oreja es un rasgo biométrico muy característico de las personas debido a que su forma y cartílago es medianamente distinguible, este rasgo se puede capturar fácilmente por medio de una cámara de seguridad y a diferencia del reconocimiento de cara sufre menos problemas de luminosidad o fondo. En este caso la cabeza actúa como un fondo y permite capturar la oreja de forma más fiable, además, es un rasgo biométrico que se mantiene estable en el tiempo, pero que muchas veces se ve afectado por estar oculto bajo el cabello o algún accesorio sobre la cabeza; normalmente se utiliza como un complemento pasivo del reconocimiento de caras o de la forma de andar. (Serratosa, 2008) (Portilla Jiménez, 2018)

Figura 10 Oreja

Oreja

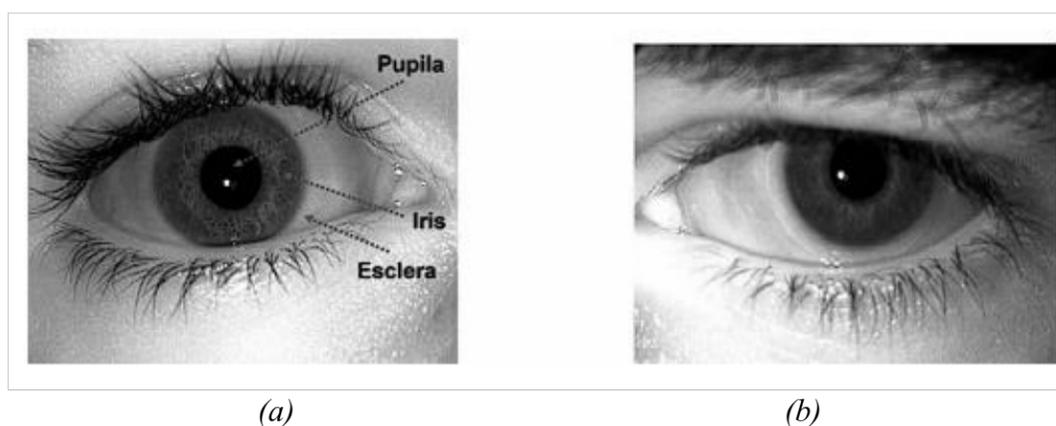


Nota. Datos tomados de (Serratosa, 2008).

5.1.6.1.4 Iris. “El iris es la región anular del ojo que se encuentra entre la pupila (el círculo negro central) y la esclera (la parte blanca externa)” (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008), su textura visual “se determina por el proceso caótico y morfogenético durante el desarrollo embrional”. (Serratos, 2008) El nivel de complejidad e información que presenta la textura del iris, permite que sea un rasgo biométrico muy distintivo para cada persona y cada ojo. Ante la comunidad científica el iris representa el rasgo biométrico más identificativo a excepción del ADN y su captura no requiere de contacto físico con el sensor.

Figura 11 (a) Estructura del ojo – (b) Ojo ocultado por el parpado

(a) Estructura del ojo – (b) Ojo ocultado por el parpado



Nota. Datos tomados de (Serratos, 2008).

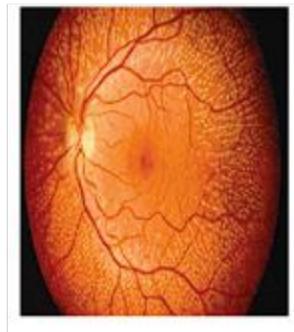
Uno de los inconvenientes que presenta el iris es que tiene un tamaño muy pequeño, por lo que se requiere de la colaboración por parte del usuario para realizar la captura, (aunque existen algunos prototipos de laboratorio que permiten la captura del iris sin dicha necesidad), el usuario se posiciona frente al dispositivo a una distancia predeterminada del plano focal de la cámara y se captura la foto, por otra parte, estos sensores suelen ser costosos y el uso de gafas o

lentes de contacto afectan en el rendimiento del sistema, haciendo necesario que la persona se los quite, además, en muchas ocasiones la morfología de las personas hace que parte del iris se vea oculto por el párpado, siendo el caso de las personas con características orientales, a pesar de esto, la tecnología del iris ha demostrado tener un nivel de precisión en imágenes de alta resolución.

5.1.6.1.5 Retina. De igual forma que el iris, la retina ha resultado ser una parte del ojo altamente discriminativa, siendo uno de los últimos rasgos biométricos utilizados en el reconocimiento de las personas. Para llevar a cabo la captura de la retina (membrana que se encuentra dentro del ojo y recibe los rayos de luz), es necesario realizar un mapeo con una luz infrarroja de los vasos sanguíneos que allí se encuentran, este proceso suele disminuir el nivel de aceptación para el reconocimiento de personas. (Serratos, 2008)

Figura 12 *Retina*

Retina



Nota. Datos tomados de (Serratos, 2008).

Según el rasgo biométrico de la cara, se pueden extraer las características que cumplan con el requerimiento para la verificación o identificación del sistema:

Tabla 1 Bondad de los rasgos biométricos de la cabeza

Bondad de los rasgos biométricos de la cabeza

Rasgo biométrico	Característica						
	Universalidad	Particularidad	Permanencia	Medible	Rendimiento	Aceptabilidad	No falsificable
Cara	A	B	M	A	B	A	A
Termografía facial	A	A	B	A	M	A	A
Oreja	M	M	A	M	M	A	M
Iris	A	A	A	M	A	B	A
Retina	A	A	M	B	A	B	A

Nota. Las entradas de la tabla muestran la bondad de cada característica en cada rasgo biométrico discretizada en tres valores: A (alto), M (medio) y B (Bajo). Datos tomados de (Serratosa, 2008)

5.1.6.2 Rasgos biométricos de la mano y los dedos.

5.1.6.2.1 Geometría de la mano y de los dedos. La mano posee características muy particulares que son invariantes, pero poco distintivas como “la forma de la mano, el tamaño de la palma y la longitud y anchura de los dedos”. (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008)

Implementar un sistema de reconocimiento basado en la geometría de la mano tiene un coste muy bajo, ya que solo es necesario tomar una fotografía de la mano de la persona posicionada sobre el sensor (en algunos sistemas se realiza una captura no solo de la palma sino también del lateral y el dorsal de la mano), además, no es necesario contar con un volumen alto

de almacenamiento, por lo que se puede implementar en sistemas de memoria limitada. También, este tipo de sistemas no se ven afectados por factores externos y es aceptado por los usuarios debido a su practicidad de hágalo usted mismo.

A pesar de sus beneficios, la geometría de la mano no presenta mucha discriminación y puede variar según el crecimiento de la persona, de igual forma, el sistema se puede ver afectado por el uso de artículos de joyería, enfermedades como la artritis o por la mutilación de uno o varios de los dedos o de la mano por completo y por el rechazo de algunas personas por cuestiones de higiene, debido a esto, se implementa en sistemas de verificación, siendo poco o nada escalable en sistemas de identificación.

Figura 13 Geometría de la mano

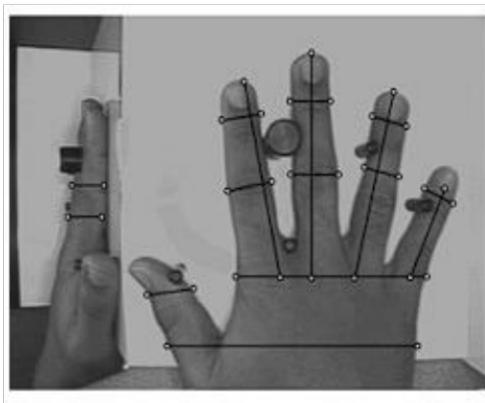
Geometría de la mano



Nota. Datos tomados de (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008)

Figura 14 Geometría de los dedos de la mano

Geometría de los dedos de la mano



Nota. Datos tomados de (Serratosa, 2008).

5.1.6.2 Huella dactilar. La huella dactilar ha sido uno de los métodos de identificación de personas que más ha trascendido a lo largo de la historia, “desde hace más de un siglo, se ha demostrado que es una tecnología altamente discriminadora incluso basándose en datos de más de 50 millones de usuarios” (Serratosa, 2008), se caracteriza por tener una serie de patrones de valles y cordilleras e información determinados en los primeros meses de gestación que permanecen hasta la descomposición de la persona, por otra parte, “la sudoración, la secreción sebácea y la suciedad de la piel hacen que el contacto del dedo con casi cualquier superficie (metal, cristal, plástico, madera, etc.) produzca en la misma una huella latente que puede ser posteriormente capturada”. (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008)

Figura 15 (a) Fin de cresta - (b) Bifurcación de cresta

(a) Fin de cresta - (b) Bifurcación de cresta



Nota. Datos tomados de (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008).

“Tradicionalmente la huella se ha capturado impregnando el dedo en tinta y haciéndolo rodar en un papel, por lo que para su uso en sistemas automáticos es preciso primero digitalizar las huellas con un escáner de documentos” (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008) que genera una imagen digital. “Hoy en día, el reconocimiento de las huellas dactilares es una tecnología muy fácil de instalar y barata” (Serratosa, 2008), el bajo costo de los sensores ha permitido que se pueda integrar con diferentes tecnologías, siendo implementada en dispositivos portátiles como laptops, celulares, etc.

A pesar de su practicidad la huella puede presentar inconvenientes que la hacen inadecuada para algunas personas, siendo el caso de las personas que realizan labores manuales o personas que tienen quemaduras o amputación de dedos. De igual forma, en algunas condiciones, la calidad de la imagen digital se puede ver afectada por la humedad o suciedad del dedo o del sensor y por el rechazo de algunas personas debido a temas de higiene, ya que el sensor es de uso colectivo.

Figura 16 (a) Ejemplo de huella - (b) Ejemplo de huella de baja calidad

(a) Ejemplo de huella - (b) Ejemplo de huella de baja calidad



Nota. Datos tomados de (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008).

5.1.6.2.3 Huella de la mano. A diferencia de los dedos, la mano posee un área discriminativa de mayor extensión ya que contiene un volumen alto de información de crestas y valles. La desventaja de usar este tipo de rasgo biométrico es que los sensores son más grandes, por consecuencia son más caros y son inutilizables en aplicaciones que implementan dispositivos pequeños, sin embargo, la ventaja de capturar la palma de la mano es que se pueden visibilizar las líneas y marcas que la componen de forma más óptima y pueden ser captadas con dispositivos de baja resolución.

Figura 17 Huella de la mano

Huella de la mano

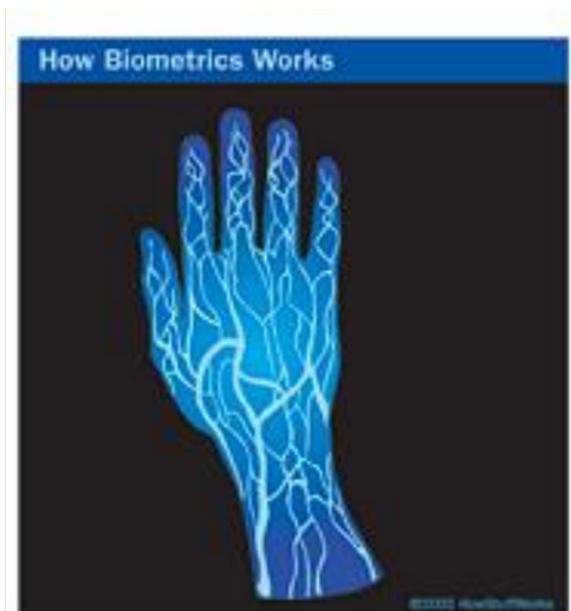


Nota. Datos tomados de (Serratos, 2008)

5.1.6.2.4 Venas de la mano y de los dedos. Las venas de las manos y los dedos poseen patrones discriminatorios que permiten verificar la identidad de las personas, este rasgo biométrico se extrae por medio del uso de sensores infrarrojos que realizan una captura de las venas de la mano que se encuentran bajo la piel y que posteriormente son comparados con los registros de la base de datos.

Figura 18 Venas de la mano y de los dedos

Venas de la mano y de los dedos



Nota. Datos tomados de (Serratosa, 2008)

Según el rasgo biométrico de la mano, se pueden extraer las características que cumplan con el requerimiento para la verificación o identificación del sistema:

Tabla 2 Bondad de los rasgos biométricos de la mano y de los dedos

Bondad de los rasgos biométricos de la mano y de los dedos

Rasgo biométrico	Característica						
	Univ ers alidad	Particular idad	Permane ncia	Medi ble	Rendimi ento	Aceptabili dad	No falsifica ble
Geometría de la mano y de los dedos	M	M	M	A	M	M	M
Dedo	M	A	A	M	A	M	M
Huella de la mano	M	A	A	B	A	M	M
Venas de la mano y de los dedos	M	M	M	M	M	M	A

Nota. Las entradas de la tabla muestran la bondad de cada característica en cada rasgo biométrico discretizada en tres valores: A (alto), M (medio) y B (Bajo). Datos tomados de (Serratosa, 2008)

5.1.6.3 Rasgos biométricos de todo el cuerpo.

5.1.6.3.1 Olor. Cada persona posee un olor característico, definido por su composición química y segregado a través de las glándulas sudoríparas, condición que permite la identificación de las personas. En épocas anteriores ya se conocía de esta característica, por lo que se adiestraba perros, para olfatear y reconocer personas que estuvieran buscando por algún motivo. En la actualidad existen mecanismos electrónicos que permiten realizar una identificación de los olores expelidos en el aire a través de una matriz de sensores de identificación de sustancias químicas o componentes aromáticos. Este tipo de sistemas está basado en la concentración de determinada sustancia química detectada por los sensores, una vez realizado el proceso de extracción de olores, se introduce aire completamente limpio dentro del sistema para su inicialización.

Este tipo de tecnología se encuentra en proceso de investigación para sistemas completamente automatizados, aunque cabe resaltar que no solo se encuentran en fase experimental, sino que se ven aplicados en sistemas reales, pero que no están predispuestos para la identificación de personas. Además, el olor es una característica altamente compleja de calcular con precisión, ya que estos sistemas se ven afectados principalmente por no poder replicar de forma contundente la precisión de la nariz humana, también el olor se puede ver influenciado por fuentes externas como el estado de salud de la persona, su higiene, el uso de fragancias y demás componentes químicos que puedan alterar su olor.

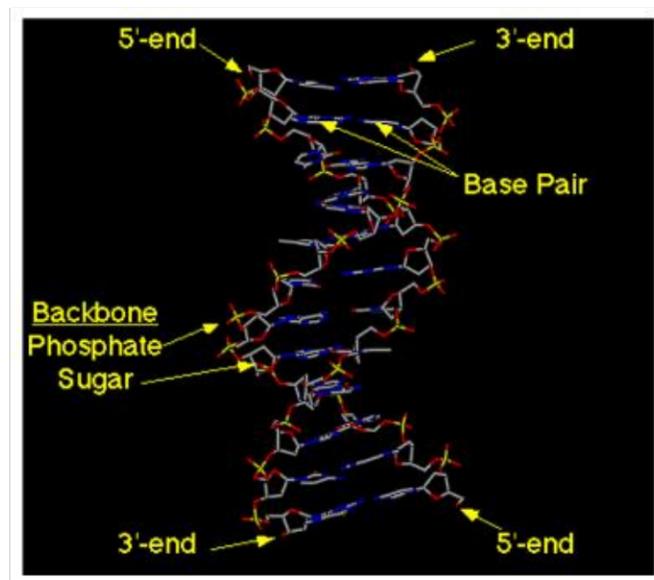
Figura 19 Olor*Olor*

Nota. Datos tomados de (Serratosa, 2008).

5.1.6.3.2 ADN. El ácido desoxirribonucleico, más conocido como ADN, es una estructura proteica compleja que constituye el material genético de los seres vivos y se encuentra dentro del núcleo de las células. El ADN es considerado por varios autores como un rasgo biométrico medianamente aceptable para ser implementado en sistemas de reconocimiento de personas, debido a que la extracción de las características del material genético requiere de horas de trabajo y de un análisis de laboratorio realizados por un experto, esta condición a pesar de que permite reconocer e identificar a las personas, hace que sea vuelva inviable para aplicaciones de sistemas automatizados que funcionan en tiempo real. Además, el estudio de la información genética del ADN de las personas vulnera su privacidad, permitiendo conocer enfermedades que padece o puede padecer. Por lo general, el estudio del ADN se ve reflejado en aplicaciones forenses y policiales.

Figura 20 ADN

ADN



Nota. Datos tomados de (Serratosa, 2008).

Según el rasgo biométrico de todo el cuerpo, se pueden extraer las características que cumplan con el requerimiento para la verificación o identificación del sistema:

Tabla 3 Bondad de los rasgos biométricos de todo el cuerpo

Bondad de los rasgos biométricos de todo el cuerpo

Rasgo biométrico	Característica						
	Universalidad	Particularidad	Permanencia	Medible	Rendimiento	Aceptabilidad	No falsificable
Olor	A	M	A	B	B	M	B
ADN	A	A	A	B	A	B	A

Nota. Las entradas de la tabla muestran la bondad de cada característica en cada rasgo biométrico discretizada en tres valores: A (alto), M (medio) y B (Bajo). Datos tomados de (Serratosa, 2008).

5.1.6.4 Rasgos biométricos de comportamiento.

5.1.6.4.1 Hablador (reconocimiento de voz). La voz es el resultado de una combinación de características fisiológicas y conductuales de cada persona, está determinada por el tamaño y forma de la cavidad tracto vocal y “se produce a través del aparato respiratorio, fonador y resonador” (Preparadores de oposiciones, 2022). Fisiológicamente la voz es estable con el tiempo y no presenta alteraciones, por el contrario, conductualmente puede ser afectada por factores como la salud, el estado de ánimo, la edad y el contexto social, haciendo que sea un rasgo poco discriminatorio.

La voz puede ser utilizada en diferentes sistemas biométricos no intrusivos y puede capturarse fácilmente a través de un micrófono o un canal telefónico, este hecho la convierte en

un rasgo biométrico comúnmente utilizado en aplicaciones de reconocimiento de voz que gestionan servicios automáticos o de interacción remota con dispositivos electrónicos. Por otra parte, existen sistemas de reconocimiento de voz basados en lectura y otros totalmente independientes. El primero, reconoce la voz de acuerdo a un texto leído por el usuario y el segundo, reconoce la voz independientemente del texto; en este tipo de sistemas el usuario suele decir cualquier palabra o frase.

Los sistemas que implementan el reconocimiento de voz pueden verse vulnerados por medio de la imitación de la voz de una persona o la reproducción de una grabación. Afrontar estas vulnerabilidades es primordial, por esta razón, este tipo de sistemas predisponen a los usuarios locuciones dinámicas que evitan el uso de grabaciones previas y no solo identifican la voz sino también la entonación, ritmo, léxico y jerga de cada persona; de igual forma la voz puede verse afectada por factores ambientales o de interferencia.

Figura 21 *Hablador (reconocimiento de voz)*

Hablador (reconocimiento de voz)



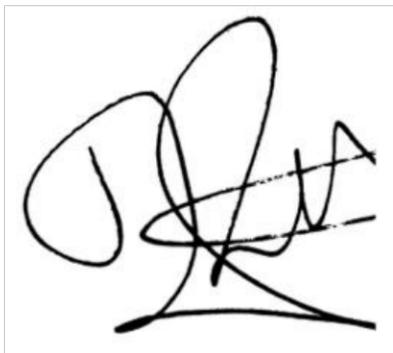
Nota. Datos tomados de (Serratosa, 2008).

5.1.6.4.2 Firma. “La firma es un mecanismo de validación de identidad usado desde hace siglos en entornos legales, gubernamentales y” transaccionales. (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008) Es un rasgo biométrico de comportamiento que puede variar con el tiempo y está determinado por factores físicos y emocionales, además, requiere de la intervención directa de la persona para ser utilizado y es susceptible a falsificaciones.

Existen dos sistemas de reconocimiento basados en la firma, el estático y el dinámico, el estático se caracteriza por que la firma se realiza sobre papel y luego se escanea para ser tratada, en cambio, en el dinámico la firma se realiza sobre una superficie capacitiva que captura el desplazamiento del lápiz y la presión a la que es sometido, en algunos casos, no es necesario que el lápiz tenga contacto con ninguna superficie. Entre ambos sistemas, el sistema dinámico es más seguro que el estático, debido a que no solo tiene como dato de entrada la firma, sino todo el proceso de su escritura, generando un conjunto de información que permite verificar la originalidad de la firma.

Figura 22 *Firma*

Firma



Nota. Datos tomados de (Serratosa, 2008).

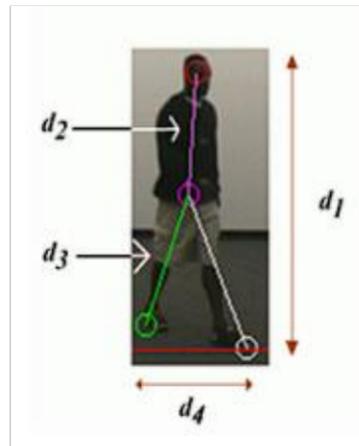
Figura 23 (a) ejemplo de firma - (b) ejemplo de imitación

(a) ejemplo de firma - (b) ejemplo de imitación



Nota. Datos tomados de (Ortega García, Alonso Fernández, & Coomonte Belmonte, 2008)

5.1.6.4.3 Forma de andar. La forma de andar es una característica muy particular de cada persona, es un rasgo biométrico de comportamiento que, a pesar de no ser muy distintivo, no es intrusivo y es fácil de capturar. La forma de andar está determinada por el ritmo y la rapidez con que se mueven las piernas al caminar, el paso y el balanceo de la persona. La persona es capturada a través de una cámara de vídeo, por lo que no es necesario que coopere para interactuar con el sistema, el algoritmo realiza una extracción de la silueta del sujeto a identificar y determina las características de los movimientos en relación al espacio tiempo. Este rasgo biométrico presenta una serie de desventajas que lo convierten en un rasgo biométrico complementario, debido a que se ve afectado por alteraciones como: la carga de algún objeto pesado, el cansancio, el tipo de vestimenta, las enfermedades, los cambios de peso, el estado físico, entre otros.

Figura 24 Forma de andar*Forma de andar*

Nota. Datos tomados de (Serratosa, 2008).

5.1.6.4.4 Manera de teclear (Dinámica de tecleo). Todas las personas tienen una forma específica de teclear que se repite consecutivamente, variando entre las pulsaciones de cada tecla o el sostenimiento de la tecla. Este rasgo biométrico es un rasgo complementario que sirve para la verificación de una persona, determinando que probablemente sea o no la persona que está tecleando. No es un rasgo biométrico altamente discriminatorio debido a que es una característica del comportamiento. Normalmente se captura de forma encubierta, capturando las pulsaciones por medio de un software instalado en el computador, es suficiente con un uso poco prolongado para descubrir si la dinámica de tecleo es el de la misma persona o no, si se observan cambios, se determina que es otra persona que está tecleando y se bloquea el sistema.

Figura 25 Manera de teclear (Dinámica de tecleo)*Manera de teclear (Dinámica de tecleo)*



Nota. Datos tomados de (Serratosa, 2008).

Según el rasgo biométrico del comportamiento, se pueden extraer las características que cumplan con el requerimiento para la verificación o identificación del sistema:

Tabla 4 Bondad de los rasgos biométricos de comportamiento

Bondad de los rasgos biométricos de comportamiento

Rasgo biométrico	Característica						
	Universalidad	Particularidad	Permanencia	Medible	Rendimiento	Aceptabilidad	No falsificable
Hablador	M	B	B	M	B	A	B
Firma	B	B	B	A	B	A	B
Forma de andar	M	B	B	A	B	A	M
Forma de teclear	B	B	B	M	B	M	M

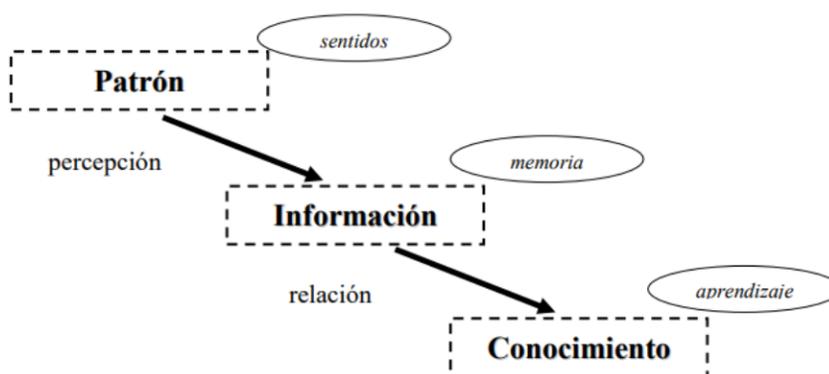
Nota. Las entradas de la tabla muestran la bondad de cada característica en cada rasgo biométrico discretizada en tres valores: A (alto), M (medio) y B (Bajo). Datos tomados de (Serratosa, 2008).

5.1.7 Reconocimiento de Patrones

El reconocimiento de patrones es un mecanismo cognitivo del sistema de visión humano, de manera instintiva, se podría definir como la forma de interpretar al mundo, normalmente la información que percibimos se encuentra codificada en patrones complejos, por ejemplo: la identificación de sonidos, sabores, olores, formas, etc. El proceso que lleva a cabo el cerebro para *reconocer patrones* se conoce de manera general, sin embargo, sigue siendo un campo abierto a la investigación. (Razo Gil, 2009)

Figura 26 Proceso biológico de reconocimiento de patrones para un ser humano

Proceso biológico de reconocimiento de patrones para un ser humano



Nota. Datos tomados de (Razo Gil, 2009)

Los *patrones* que percibimos en nuestra vida diaria son adquiridos por medio de nuestros órganos sensoriales, por ejemplo, una persona, aprende a reconocer su entorno identificando patrones visuales, auditivos y sensitivos, para asociar estos nuevos patrones con unos idénticos o parecidos es necesario recurrir a *información* previa que relacione y establezca una correspondencia entre la percepción actual y la recordada, de tal manera que conforme crece, la persona, va refinando su capacidad de reconocimiento a través de la asignación de etiquetas,

logrando así distinguir una melodía de otra, diferentes variaciones de temperatura o diferentes tonalidades de un mismo color. Finalmente, a partir de la acumulación de la información adquirida (*conocimiento*) surge el *aprendizaje*, que frecuentemente genera experiencias que producen cambios en la conducta de la persona.

Con el crecimiento de los avances tecnológicos y curiosidad por imitar las habilidades del sistema biológico del ser humano, el hombre se ha permitido desarrollar sistemas computacionales capaces de replicar el comportamiento humano, abriendo paso al surgimiento de nuevas áreas de investigación como la *inteligencia artificial*. Los sistemas expertos, el razonamiento automático, el procesamiento del lenguaje natural y el reconocimiento de patrones o visión computacional, son algunas de las tareas que abarca la inteligencia artificial, siendo esta última tarea, una valiosa herramienta en estos sistemas. (Razo Gil, 2009)

El reconocimiento de patrones, también se define como “la ciencia que se ocupa de los procesos sobre ingeniería, computación y matemáticas relacionados con objetos físicos y/o abstractos, con el propósito de extraer información que permita establecer propiedades de o entre conjuntos de dichos objetos, los cuales nos permiten interpretar el mundo que nos rodea”. Los fundamentos de este campo se establecieron ya en los años sesenta y setenta, con excelentes libros (Duda y Hart, 1973; Fukunaga, 1972), que dieron forma a este campo de investigación. (Alegre Gutiérrez, Pajares Martinsanz, & de la Escalera Hueso, 2016)

La tarea de reconocimiento de patrones implica necesariamente un proceso de aprendizaje a partir de un conjunto de objetos (*conjunto de datos*), donde el objetivo es (Alegre

Gutiérrez, Pajares Martinsanz, & de la Escalera Hueso, 2016) clasificar *objetos* en un cierto número de *categorías* o *clases*. Los objetos suelen ser representaciones de elementos que están sujetos a estudio. Un *patrón* suele ser la descripción de un objeto en términos de sus *características* que a su vez, ofrecen información descriptiva, cuantitativa y estructural sobre el patrón. Un conjunto de patrones con características similares representa una clase. El proceso de *clasificación* consiste en agrupar objetos en clases, según sea su semejanza o diferencia que existe entre ellos. (Razo Gil, 2009)

Normalmente, los objetos se representan a través de características observadas en ellos, en la forma vectorial:

$$x = [x_1, x_2, \dots, x_l]^T$$

Donde el vector x está formado por las $x_i, i = 1, 2, \dots, l$ características; el operador T representa la operación matricial transpuesta.

En un sentido más formal podríamos definir:

5.1.7.1 Patrón. Un patrón es la entidad que puede ser identificable de acuerdo a sus características o atributos, éste puede ser abstracto o concreto. Los patrones abstractos son ideas conceptuales mientras que un patrón concreto es la representación física de un objeto, por ejemplo, los símbolos (letras, caligráficas), imágenes digitales, imágenes tridimensionales,

firmas, señales de voz, electrocardiogramas, electroencefalograma, ondas sísmicas, entre otros.

(Vázquez López & Cuevas de la Rosa , 2017)

5.1.7.2 Clase. Una clase es un estado o categoría de los patrones. Los patrones que pertenecen a la misma clase comparten los mismos atributos y comportamientos. En un sentido abstracto, una clase, se puede interpretar como un modelo o prototipo que idealiza un objeto. Matemáticamente representamos una clase i -ésima con el símbolo ω_i , donde ω_i . El conjunto de todas las clases se le conoce como espacio de interpretación. (Vázquez López & Cuevas de la Rosa , 2017)

5.1.7.3 Características. Las características describen a los patrones, de acuerdo al valor de sus atributos, y determinan el espacio de características de una clase. En la siguiente tabla se presenta una lista de las características comúnmente utilizadas en el reconocimiento de señales e imágenes. (Vázquez López & Cuevas de la Rosa , 2017)

Tabla 5 Características comunes en el reconocimiento de señales

Características comunes en el reconocimiento de señales

Atributos de señales	Atributos de imágenes
Amplitud	Área de región
Medidas del histograma	Color
Desviación estándar	Intensidad
Morfología de la onda	Momentos invariantes
Cruce por ceros	Histograma
Correlación	Análisis de Fourier
Espectros de frecuencias	Morfología de la imagen

Nota. Datos tomados de (Vázquez López & Cuevas de la Rosa , 2017).

Los atributos pueden tener valores de tipo numérico, cadenas alfanuméricas o símbolos. Sí éstos son de tipo numérico, entonces pueden ser representados mediante vectores. Por ejemplo, la ecuación presentada, el vector x representa un vector de características con d características. (Vázquez López & Cuevas de la Rosa , 2017)

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}$$

Sí los valores son caracteres o símbolos pueden representarse mediante cadenas o una estructura de árbol. Por ejemplo, el ADN, está constituido por una cadena de proteínas: adenina (A), guanina (G), citosina (C) y tiamina (T), entonces un gen puede ser descrito por una cadena como la siguiente:

GAAGTCCAG

5.1.7.4 Similitud. La noción de similitud, es la pieza angular en el proceso del RP, se refiere a los valores parecidos de un atributo en dos o más objetos. En el proceso de reconocimiento, se evalúa la similitud entre un objeto y un modelo (prototipo) que idealiza a la clase a la que pertenece. (Vázquez López & Cuevas de la Rosa , 2017)

Básicamente existen tres enfoques para evaluar la similitud de los patrones:

1) La clasificación

- 2) El análisis de regresión
- 3) La descripción

5.1.7.4.1 Clasificación. La clasificación es el proceso de asignar un patrón desconocido a una clase con base a la evaluación de sus atributos. La idea básica es reconocer objetos utilizando rasgos. Por ejemplo, para diferenciar una naranja de una manzana se pueden considerar dos atributos: la forma y el color, por lo tanto, el vector de características estaría definido por la siguiente ecuación. (Vázquez López & Cuevas de la Rosa , 2017)

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} color \\ forma \end{bmatrix}$$

La salida de este proceso es meramente una decisión sobre la clase a la que pertenece el objeto. Cada objeto es reconocido como perteneciente a un tipo particular y puede ser asignado a un conjunto de grupos preestablecidos (clases), que representan todos los tipos de objetos que se espera que existan. Ocurre un error de clasificación si se realiza la asignación a una clase inapropiada, la probabilidad de que esto ocurra es un radio de error de clasificación.

Generalmente para llevar a cabo este proceso se suele asignar un nivel de importancia a un atributo particular del objeto que influye en el proceso de clasificación, en el caso de la manzana y la naranja podríamos asignarle mayor importancia a la forma de la fruta ya que la forma de una manzana es diferente a la de una naranja.

5.1.7.4.2 Regresión. La regresión puede definirse como la determinación de valores de los parámetros de un modelo, a menudo bajo algún criterio de optimización. La tarea de regresión se fundamenta en el proceso cognitivo de la inferencia, es una herramienta que permite calcular el valor de una variable a partir de valores antiguos de la misma o de otras variables. La desviación entre el valor estimado y el real genera un error de aproximación. El objetivo del sistema de regresión es que el error sea lo menor posible. (Vázquez López & Cuevas de la Rosa , 2017)

5.1.7.4.3 Descripción. La descripción se basa en el lenguaje formal, busca medir la similitud de los objetos a partir de una regla estructural la cual está conformada de una secuencia de primitivas. Tanto la clasificación como la regresión evalúan la similitud realizando una medición de distancias; mientras que la descripción realiza la evaluación a partir de reglas sintácticas previamente definidas. (Vázquez López & Cuevas de la Rosa , 2017)

5.1.7.5 Enfoques del reconocimiento de patrones. El reconocimiento de patrones es capaz de distinguir entre diferentes tipos de ambientes, señales que permitan tomar decisiones para dar solución a un problema específico. Actualmente hay cuatro enfoques en el reconocimiento de patrones:

5.1.7.5.1 Reconocimiento estadístico de patrones. Se basa en la teoría de la probabilidad y estadística la cual tiene un conjunto de medidas numéricas con distribuciones de probabilidad conocidas y a partir de ellas se lleva a cabo el reconocimiento.

5.1.7.5.2 Reconocimiento sintáctico de patrones. Encuentra relaciones estructurales que guardan los objetos de estudio, empleando la teoría de lenguajes formales, la idea esencial es el desarrollo de una gramática que determine la estructura de dichos objetos.

5.1.7.5.3 Redes neuronales. Se determina una estructura de neuronas interconectadas que se estimulan unas a otras, estas son entrenadas para dar respuesta cuando se presentan una serie de valores.

5.1.7.5.4 Reconocimiento lógico combinatorio de patrones. Se encarga de modelar el problema lo más cercano posible a la realidad. Las características usadas para describir los objetos de estudio deben ser definidas cuidadosamente.

5.1.8 Técnicas de reconocimiento de patrones

Existen diversos algoritmos para construir clasificadores de reconocimiento de patrones, que suelen agruparse de la siguiente manera:

5.1.8.1 Estimación paramétrica. Se basa en la aproximación de la probabilidad condicional de que un patrón pertenezca a una clase, asumiendo alguna forma para dicha probabilidad y suponiendo el número de clases presentes. “El propósito de estos métodos es obtener una hipótesis que permita clasificar a nuevos individuos” (Gómez Gil & Sucar Succar, 2019). Los algoritmos principales son:

5.1.8.1.1 Clasificador de Bayes (Naïve Bayes). El enfoque “bayesiano” es usado para el reconocimiento de objetos especialmente cuando la distribución de los objetos no es tan directa y se desea tomar una buena decisión. En general, si existe una superposición significativa en los valores de rasgos de diferentes objetos, es posible que se encuentren varias clases como potenciales candidatas a las que se puede asignar el mismo. (Sanz, 2022)

El clasificador de Naïve Bayes (NB), está basado en el teorema de Bayes y propone una hipótesis de independencia entre las características o variables predictoras; debido a esta hipótesis recibe el calificativo de Naïve que significa “ingenuo” (Duda y cols., 2000). El teorema de Bayes da la probabilidad de una clase C dado un conjunto de características x_1, \dots, x_n como:

$$P(C|x_1, \dots, x_n) = \frac{P(C)P(x_1, \dots, x_n|C)}{P(x_1, \dots, x_n)}$$

Sustituyendo el numerador por su representación como probabilidad condicional, y asumiendo que $P(x_i|C, x_j) = P(x_i|C)$, esta ecuación puede escribirse como:

$$P(C|x_1, \dots, x_n) = \frac{P(C) \prod_{i=1}^n P(x_i|C)}{P(x_1, \dots, x_n)}$$

Este algoritmo utiliza una regla de clasificación en función de la clase o hipótesis más probable, conocido como “máximo a posteriori” (MAP). La función de clasificación puede expresarse como (Mitchell, 1997):

$$\hat{C} = \arg \max_{c \in C} P(c) \prod_{i=1}^n P(x_i|c)$$

Diferentes modelos de clasificación NB pueden crearse cambiando la función de distribución aplicada sobre $P(x_i|c)$; al trabajar con datos continuos se puede asumir una distribución normal, mientras que cuando los datos son discretos es más común utilizar una distribución multinomial o de Bernoulli. (Sanz, 2022)

La gran ventaja de NB es que puede entrenarse usando pocos datos y es capaz de tolerar características irrelevantes. Sin embargo, la suposición de independencia entre las características es algo que no siempre se presenta, puesto que en la mayoría de los problemas existe una correlación entre las mismas. Aun así, puede conseguir un buen desempeño. (Sanz, 2022)

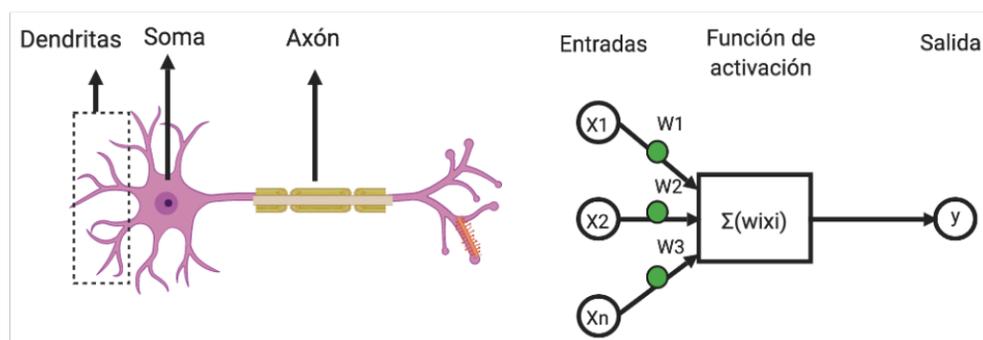
5.1.8.1.2 Redes neuronales. Una red neuronal es “un nuevo sistema para el tratamiento de la información, cuya unidad básica de procesamiento está inspirada en la célula fundamental del sistema nervioso humano: la neurona” (Ruiz, Basualdo, & Matich, 2001). La mayoría de procesos del cuerpo humano se relacionan de una forma u otra con la actividad neuronal. Las neuronas son un componente simple, que en conjunto se hacen poderosas. Básicamente lo que ocurre con una neurona es lo siguiente: las neuronas son estimuladas a través de sus *entradas*(inputs) y cuando logran llegar a cierto umbral se disparan o activan, enviando una señal hacia el *axón*.

Partiendo de lo anterior, las redes neuronales artificiales podrían definirse como modelos matemáticos basados en el comportamiento del sistema nervioso, por consiguiente y considerando lo explicado anteriormente, para una red neuronal artificial la unidad fundamental de procesamiento de información son las neuronas artificiales.

En ese sentido, podríamos decir que las redes neuronales artificiales son un conjunto de unidades de procesamiento llamadas neuronas, células o nodos, interconectados entre sí por varios enlaces llamados conexiones, con el objetivo de recibir señales de entrada, procesarlas y enviar una respuesta (salida). Como se puede observar en la Figura 27, se compara una neurona biológica con una neurona artificial y se pueden observar similitudes como las entradas, además cada enlace tiene asociado un peso específico que representa la información utilizada por los nodos para resolver un problema. (Ruiz, Basualdo, & Matich, 2001)

Figura 27

Representación comparativa de una red neuronal



Nota. Datos tomados de (Vega Huerta, et al, 2009).

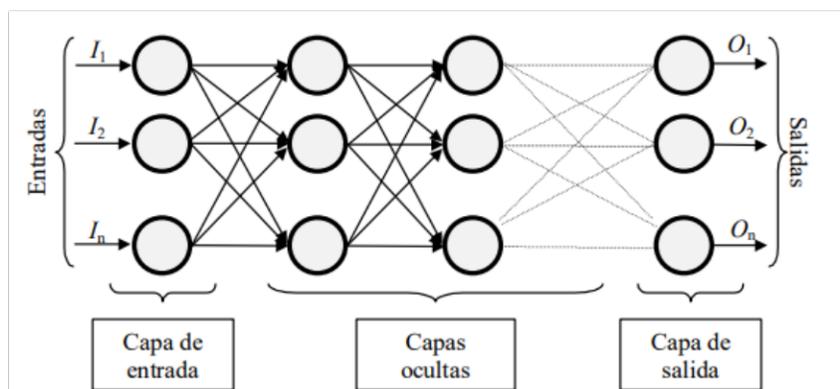
Las entradas en una neurona pueden provenir de otras neuronas o fuentes externas, de la misma forma, las salidas de una neurona pueden ser enviadas al entorno o ser entradas de otras

neuronas. Para lograr resolver un problema específico, el conocimiento en una red neuronal se distribuye en todo el sistema a través de muchas neuronas interconectadas. (Vega Huerta, et al, 2009).

A continuación, podemos observar un esquema de red neuronal (Figura 28 RNA totalmente conectada), al igual que una neurona biológica, una neurona artificial, tiene varias entradas y una o varias salidas, que a su vez puede convertirse en entrada a muchas otras neuronas.

Figura 28

RNA totalmente conectada



Nota. Datos tomados de (Paz Arias & Jimenez, 2018).

Normalmente, las RNA están constituidas por neuronas interconectadas y distribuidas en tres capas (esto último puede variar). Los datos ingresan a través de la “*capa de entrada*”, luego pasan por la “*capa oculta o de activación*”, la cual puede estar conformada por varias capas y finalmente los datos salen por la “*capa de salida*”. (Paz Arias & Jimenez, 2018)

Durante la **función de entrada (input function)**, la neurona interpreta a muchos valores de entrada como si fueran uno solo; esto se conoce como *entrada global*. Por lo tanto nos enfrentamos al problema de cómo combinar las entradas simples (in_{i1}, in_{i2}, \dots) dentro de la entrada global gin_i . Esto se logra a través de la función de *entrada*, la cual se calcula a partir del *vector entrada*. La función de entrada puede describirse de la siguiente manera:

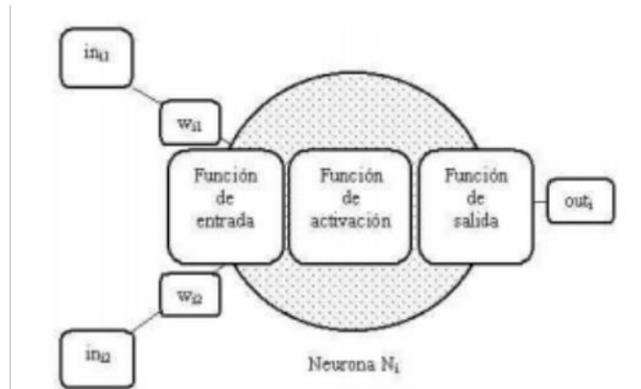
$$input_i = (in_{i1} \cdot w_{i1}) * (in_{i2} \cdot w_{i2}) * \dots (in_{in} \cdot w_{in})$$

donde: * representa al operador apropiado (por ejemplo: máximo, sumatoria, productoria, etc.), n al número de entradas a la neurona N_i y w_i al peso. (Paz Arias & Jimenez, 2018)

Los valores de entrada se multiplican por los pesos previamente ingresados a la neurona. Los pesos que generalmente no están restringidos, cambian la medida de influencia que tienen los valores de entrada. Es decir, que permiten que un gran valor de entrada tenga solamente una pequeña influencia, si estos son lo suficientemente pequeños. (Paz Arias & Jimenez, 2018)

Figura 29

Ejemplo de una neurona con 2 entradas y 1 salida



Nota. Datos tomados de (Ruiz, Basualdo, & Matich, 2001).

La nomenclatura utilizada en la Figura 29 es la siguiente: in_{i1} = entrada número 1 a la neurona N_i ; w_{i1} = peso correspondiente a in_{i1} ; in_{i2} = entrada número 2 a la neurona N_i ; w_{i2} = peso correspondiente a in_{i2} ; y out_i = salida de la neurona N_i . El conjunto de todas las entradas $in_i = (in_{i1}, in_{i2}, \dots, in_{in})$ es comúnmente llamado “vector entrada”. (Ruiz, Basualdo, & Matich, 2001)

Algunas de las funciones de entrada más comúnmente utilizadas y conocidas son:

- La *sumatoria de las entradas pesadas*, es la suma de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos.

$$\sum_j (in_{ij} w_{ij}), \text{ con } j = 1, 2, \dots, n$$

- La *productoria de las entradas pesadas*, es el producto de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos.

$$\prod_j(n_{ij}w_{ij}), \text{ con } j = 1, 2, \dots, n$$

- El *máximo de las entradas pesadas*, solamente toma en consideración el valor de entrada más fuerte, previamente multiplicado por su peso correspondiente.

$$\text{Max}_j(n_{ij}w_{ij}), \text{ con } j = 1, 2, \dots, n$$

Una vez finalizado el proceso de entrada, en la *función de activación (activation function)*, una neurona biológica puede estar activa (excitada) o inactiva (no excitada); es decir, que tiene un “estado de activación”. Las neuronas artificiales también tienen diferentes estados de activación; algunas de ellas solamente dos, al igual que las biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado. (Ruiz, Basualdo, & Matich, 2001)

La función activación calcula el estado de actividad de una neurona; transformando la entrada global (menos el umbral Θ_i) en un valor (estado) de activación, cuyo rango normalmente va de (0 a 1) o de (-1 a 1). Esto es así, porque una neurona puede estar totalmente inactiva (0 o -1) o activa (1).

La función activación, es una función de la entrada global (gin_i) menos el umbral (Θ_i). Las funciones de activación más comúnmente utilizadas se detallan a continuación:

- En la *función lineal*, los valores de salida obtenidos por medio de esta función de activación serán: $a \cdot (gin_i - \Theta_i)$ cuando el argumento de ($gin_i - \Theta_i$) esté comprendido

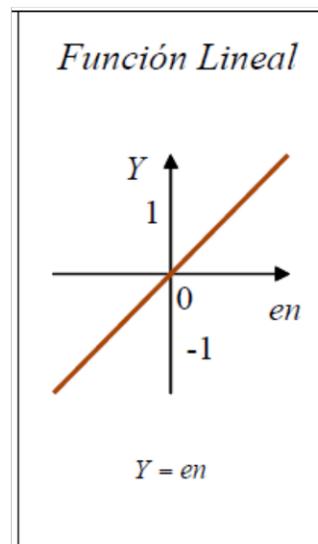
dentro del rango $(-1/a, 1/a)$. Por encima o por debajo de esta zona se fija la salida en 1 o -1 , respectivamente. Cuando $a = 1$ (siendo que la misma afecta la pendiente de la gráfica), la salida es igual a la entrada. (Ruiz, Basualdo, & Matich, 2001)

$$f(x) = \begin{cases} -1, & x \leq -\frac{1}{a} \\ a * x, & -\frac{1}{a} < x < \frac{1}{a} \\ 1, & x \geq \frac{1}{a} \end{cases}$$

con $x = \text{gin}_i - \Theta_i$, y $a > 0$.

Figura 30

Función de activación lineal



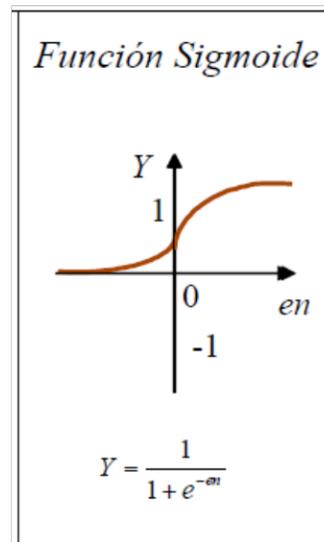
Nota. Datos tomados de (Ruiz, Basualdo, & Matich, 2001).

- En la *función sigmoidea*, los valores de salida que proporciona esta función están comprendidos dentro de un rango que va de 0 a 1. Al modificar el valor de g se ve afectada la pendiente de la función de activación. (Ruiz, Basualdo, & Matich, 2001)

$$f(x) = \frac{1}{1+e^{-gx}}, \text{ con } x = |gin_i - \Theta_i|.$$

Figura 31

Función de activación sigmoidea



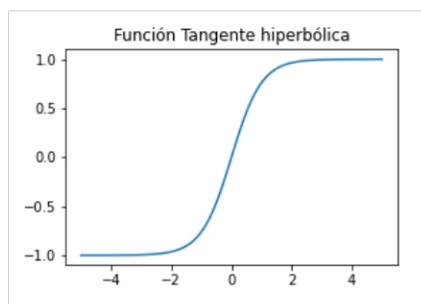
Nota. Datos tomados de (Ruiz, Basualdo, & Matich, 2001).

- En la *función tangente hiperbólica*, los valores de salida de la función tangente hiperbólica están comprendidos dentro de un rango que va de -1 a 1. Al modificar el valor de g se ve afectada la pendiente de la función de activación. (Ruiz, Basualdo, & Matich, 2001)

$$f(x) = \frac{e^{gx} - e^{-gx}}{e^{gx} + e^{-gx}}, \text{ con } x = \overline{gin}_i - \Theta_i.$$

Figura 32

Función de activación tangente hiperbólica



Nota. Datos tomados de (Ruiz, Basualdo, & Matich, 2001).

Para explicar por qué se utilizan estas funciones de activación se suele emplear la analogía a la aceleración de un automóvil. Cuando un auto inicia su movimiento necesita una cantidad de potencia considerable para comenzar a acelerar. Sin embargo, al ir tomando velocidad, comienza a utilizar una menor cantidad de potencia que la inicial para mantener la aceleración. Al alcanzar altas velocidades, el auto sufre nuevamente un incremento de potencia para obtener una pequeña ganancia de velocidad. En resumen, en ambos extremos del rango de aceleración de un automóvil se demanda una mayor potencia para la aceleración que en la mitad de dicho rango. (Ruiz, Basualdo, & Matich, 2001)

El último componente que una neurona necesita es la **función de salida (output function)**. El resultado de esta función es la salida de la neurona i (out_i); por ende, la función de salida determina que valor se transfiere a las neuronas vinculadas. Si la función de activación está por debajo de un umbral determinado, ninguna salida se pasa a la neurona subsiguiente.

Normalmente, no cualquier valor es permitido como una entrada para una neurona, por lo tanto, los valores de salida están comprendidos en el rango $[0, 1]$ o $[-1, 1]$. También pueden ser binarios $\{0, 1\}$ o $\{-1, 1\}$. (Ruiz, Basualdo, & Matich, 2001)

Dos de las funciones de salida más comunes son:

1. *Ninguna*, este es el tipo de función más sencillo, tal que la salida es la misma que la entrada. Es también llamada función identidad.
2. *Binaria*:

$$\text{Binaria: } \begin{cases} 1, & \text{si } act_i \geq \xi_i \\ 0, & \text{de lo contrario} \end{cases}, \text{ donde } \xi_i \text{ es el umbral.}$$

Las RNA además de presentar similitudes con el cerebro humano, poseen una serie de características propias, por ejemplo, las RNA aprenden de la experiencia, generalizan información basada en ejemplos y abstraen las principales características de un conjunto de datos. (Basogain Olabe)

1. **Aprender:** Las RNA pueden cambiar su comportamiento en función de su entorno. A partir de las entradas ingresadas, ellas mismas se ajustan para producir la salida deseada.
2. **Generalizar:** Adquirir el conocimiento de un objeto específico por medio del estudio. Las RNA generalizan la información automáticamente debido a su propia estructura y naturaleza. Estas pueden producir dentro de un rango específico, respuestas correctas a entradas que poseen variaciones provenientes de ruido o distorsión.

3. **Abstraer:** Las RNA son capaces de abstraer la esencia de un conjunto de entradas o datos que aparentemente no presentan aspectos similares.

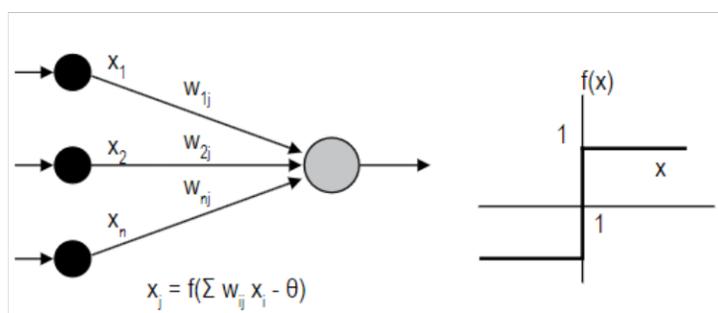
Dentro de los principales **modelos de redes neuronales artificiales** encontramos:

Perceptrón, el perceptrón está conformado por un conjunto de neuronas lineales cuyo propósito es recibir las entradas a la red y una neurona de salida. Este modelo puede determinar cuándo una entrada presentada a la red pertenece a una de las dos clases que es capaz de reconocer. A continuación, en la Figura 33

Arquitectura del perceptrón podemos apreciar la arquitectura del perceptrón. (Vega Huerta, et al, 2009)

Figura 33

Arquitectura del perceptrón



Nota. Datos tomados de (Vega Huerta, et al, 2009).

La arquitectura del Perceptrón, también es conocida como mapeo de patrones (pattern-mapping), esta arquitectura aprende a clasificar modelos mediante un aprendizaje supervisado.

Los modelos que clasifican suelen ser generalmente vectores con valores binarios (0,1) y las categorías de la clasificación se expresan mediante vectores binarios. (Vega Huerta, et al, 2009)

La única unidad de procesamiento de salida del Perceptrón realiza la suma ponderada de las entradas con base a la siguiente ecuación:

$$S_j = \sum a_i w_{ji}$$

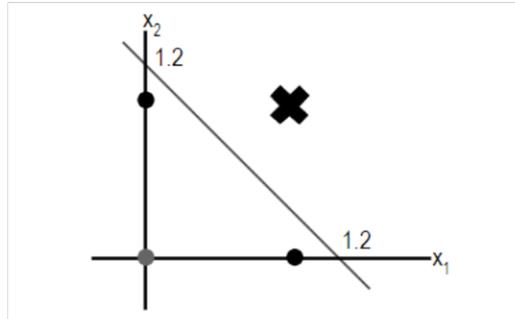
Posteriormente, la unidad de procesamiento (neurona de salida) resta el umbral y pasa el resultado a una función de transferencia de tipo escalón. La regla de decisión es responder +1 si el patrón presentado pertenece a la clase *A* y con -1 si el patrón pertenece a la clase *B*. La salida dependerá de la entrada neta (suma de las entradas X_i ponderadas) y del valor umbral 0. (Vega Huerta, et al, 2009)

El modelo del perceptrón sólo es apto para discriminar patrones sencillos, linealmente separables. La separabilidad lineal limita a las redes con sólo dos capas a la resolución de problemas en los cuáles el conjunto de puntos (correspondientes a los valores de entrada) sean separables geoméricamente. En el caso de dos entradas, la separación se lleva a cabo mediante una línea recta, tal como puede apreciarse en la Figura 34

Problema linealmente separable: Función lógica OR. Para tres entradas, la separación se realiza mediante un plano en el espacio tridimensional, y así sucesivamente hasta el caso de N entradas, en el cual el espacio N -dimensional es dividido en un hiperplano. (Vega Huerta, et al, 2009)

Figura 34

Problema linealmente separable: Función lógica OR



Nota. Datos tomados de (Vega Huerta, et al, 2009).

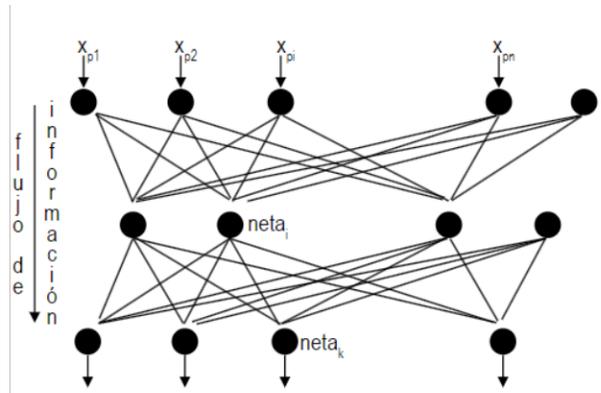
BackPropagation (Retro propagación), es un modelo de entrenamiento de redes multicapa, su potencia reside en su capacidad de entrenar capas ocultas y de este modo supera las posibilidades restringidas de las redes de una única capa. La principal característica de este algoritmo es la representación interna del conocimiento, el cual se organiza en la capa o capas intermedias, para obtener cualquier coincidencia posible entre la entrada y la salida de la red. (Vega Huerta, et al, 2009)

En la Figura 35

Modelo .Se puede apreciar la arquitectura de una red neuronal artificial modelo BackPropagation.

Figura 35

Modelo Retro propagación



Nota. Datos tomados de (Vega Huerta, et al, 2009).

El modelo de retro propagación es capaz de autoajustar los pesos de las neuronas de las capas intermedias para aprender la relación entre el conjunto de patrones dados como entrada y sus respectivas salidas, con el objetivo de reutilizar el mismo mecanismo de relación en nuevos vectores de entrada con ruido o incompletos, para obtener una salida activa si la nueva entrada es similar a las obtenidas durante el aprendizaje. (Vega Huerta, et al, 2009)

El objetivo del **entrenamiento o aprendizaje de una RNA** es lograr que un problema específico para un conjunto de entradas produzca el resultado deseado o mínimamente consistente. Este proceso consiste en la aplicación secuencial de un conjunto de vectores de entradas para lograr autoajustar los pesos de las conexiones establecidas entre las diferentes capas de la red teniendo en cuenta un procedimiento específico. (Basogain Olabe)

- 1. Entrenamiento supervisado:** Este tipo de aprendizaje requiere que cada vector de entrada coincida con su respectivo vector de salida. Este entrenamiento consiste en presentar un vector de entrada a la red, calcular los pesos y comparar la salida de la red con la salida deseada. Luego, se obtiene el margen de error entre las salidas que a

su vez es utilizado para retroalimentar la red y cambiar los pesos de acuerdo con un algoritmo que tiende a minimizar el margen de error. (Basogain Olabe)

2. **Entrenamiento no supervisado:** Este tipo de entrenamiento se basa en modelos de aprendizaje más lógicos, ya que no requieren de un vector de salidas deseadas y por lo tanto no realiza la comparación entre las salidas reales de la red y las esperadas. El conjunto de entrenamiento consiste únicamente en vectores de entrada. El algoritmo de entrenamiento modifica los pesos de la red de tal forma que produzca salidas consistentes para posteriormente extraer las propiedades estadísticas del conjunto de valores de entrenamiento y agrupar en clases los vectores similares. (Basogain Olabe)

5.1.8.1.3 Máquinas de vectores soporte. Las máquinas de soporte vectorial o SVM (por sus siglas en inglés Support Vector Machine), hacen parte de un nuevo sistema de aprendizaje, que ha tomado fuerza en los últimos años en el desarrollo de nuevos algoritmos, estrategias de implementación y como herramienta para la identificación de sistemas no lineales, resolviendo problemas de clasificación, basándose en el principio de minimización del riesgo estructural (SRM por sus siglas en inglés "Structural Risk Minimization"), principio originado de la teoría de aprendizaje estadístico desarrollada por Vapnik. (Cuevas Alfaro, 2010)

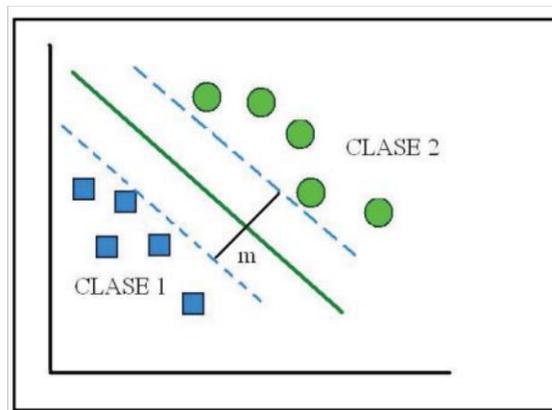
Este principio permite seleccionar un clasificador que minimiza el margen de error y proporciona una buena medida para obtener algoritmos de clasificación que integran datos desconocidos. Esto le da una ventaja a las SVM sobre otros clasificadores, como redes neuronales artificiales que pueden producir modelos que sobre ajustan los datos.

Las SVM, suponiendo que tenemos ejemplos de sólo dos categorías, construyen un hiperplano en un espacio de dimensionalidad mayor o incluso infinita. Este hiperplano separa de la mejor forma los puntos de una clase de la de otra. En el concepto de “separación óptima” es donde reside la característica fundamental de las SVM, esta consideración busca el hiperplano que tenga la máxima distancia (margen) con los puntos que estén más cerca de él mismo, como se observa en la Figura 36

Margen m . Por eso también a veces se les conoce a las SVM como clasificadores de margen máximo. (Alcalá Nalvaiz & Campo León, 2022)

Figura 36

Margen m



Nota. Datos tomados de (Cuevas Alfaro, 2010).

Las máquinas de soporte vectorial pueden contemplar dos casos de variables lineales, separables y no separables.

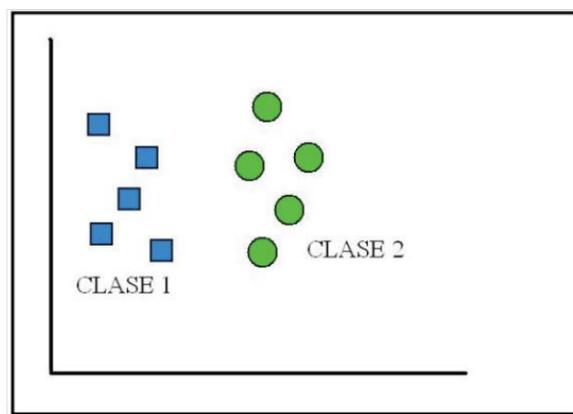
En el *caso linealmente separable*, dado un conjunto C de puntos para entrenamiento, etiquetados en dos clases denominadas $C1$ y $C2$, como se puede apreciar en la figura 2 – 2. cada elemento del conjunto C se representa:

$$(y_1, x_1), (y_2, x_2), \dots \dots \dots (y_i, x_i)$$

Los puntos de entrenamiento $x_i \in R^N$, pertenecen a alguna de las dos clases y se les ha etiquetado $y_i \in \{-1,1\}$, dado $i = 1,2, \dots, l$. Normalmente la búsqueda del hiperplano óptimo en una entrada es demasiado restrictivo para ser de uso práctico.

Figura 37

Caso separable linealmente



Nota. Datos tomados de (Cuevas Alfaro, 2010).

Para contrarrestar este inconveniente, se realiza un mapeo del espacio de entrada en un espacio con características de una mayor dimensión y se busca el hiperplano conveniente. Sea $Z = \varphi(x)$ la notación correspondiente al vector dentro del espacio de características con el mapeo φ de R^N a un espacio de características Z .

Las condiciones que se buscan dentro del hiperplano están determinadas por la formula:

$$w \cdot z + b = 0 \quad \text{E.1}$$

Este hiperplano está definido por el par (w, b) , haciendo posible separar el punto x de acuerdo con la función:

$$f(x_i) = \text{sign}(w \cdot z_i + b) = \begin{cases} 1, & y_i = 1 \\ -1, & y_i = -1 \end{cases} \quad \text{E.2}$$

donde $w \in Z$ y $b \in R$. Específicamente, el conjunto C es linealmente separable si existe (w, b) de tal forma que las inecuaciones:

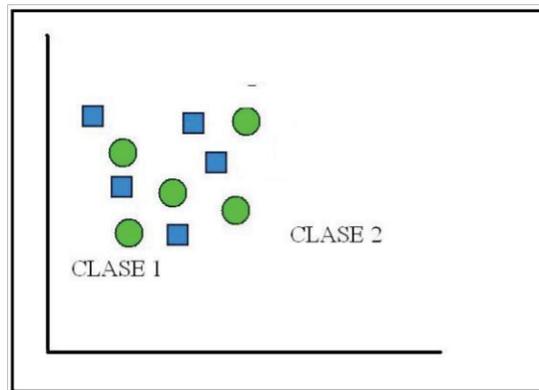
$$\begin{cases} (w \cdot z_i + b) \geq 1, & y_i = 1 \\ (w \cdot z_i + b) \leq -1, & y_i = -1 \end{cases} \quad \text{E.2}$$

Sean válidas para todos los elementos del conjunto. Para el caso linealmente separable de C , el hiperplano no está determinado por un único vector w , sino que está determinado por una infinidad de dichos vectores, a pesar de esto, solo existe un hiperplano óptimo donde el margen de proyección de los puntos de entretenimiento de dos clases tiene las condiciones de ser maximizado.

Para el *caso no linealmente separable*, dado el conjunto C con datos no linealmente separables, es necesario que se realicen alteraciones a la clasificación del planteamiento de la SVM.

Figura 38

Caso no linealmente separable



Nota. Datos tomados de (Cuevas Alfaro, 2010).

Se puede generalizar el análisis previo, introduciendo variables positivas establecidas como: $\xi_i \geq 0$, de modo que la ecuación E.2 sea modificada a:

$$y_i(w \cdot z_i + b) \geq 1 - \xi_i, i = 1, 2, \dots, l \text{ E.3}$$

Los $\xi_i \neq 0$, son aquellos donde x_i no satisface la ecuación anterior, por lo que el término: $\sum_{i=1}^l \xi_i$ puede ser considerado como un error en la clasificación.

El problema del hiperplano óptimo vuelve a ser definido entonces como la solución del problema:

$$\min \left\{ \frac{1}{2} w \cdot w + C \sum_{i=1}^l \xi_i \right.$$

$$\text{Sujeto a: } \begin{cases} y_i(w \cdot z_i + b) \geq 1 - \xi_i \\ \xi_i \geq 0 \end{cases} \quad \text{E.4}$$

Pudiendo ser C una constante definida como parámetro de regularización, cabe resaltar que C es el único parámetro que puede ser ajustado libremente dentro de la formulación del SVM, siendo ajustado como un balance entre la maximización del margen y la alteración a la clasificación.

La búsqueda del hiperplano óptimo de la función E.4 es un problema de programación cuadrática, pudiendo ser resuelto al construir un langragiano y transformándolo en dual como:

$$\text{Max } W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j z_i \cdot z_j \quad \text{E.5}$$

$$\text{Sujeto a: } \sum_{i=1}^l y_i \alpha_i = 0, 0 \leq \alpha_i \leq C, i = 1, \dots, l \quad \text{E.6}$$

Donde $\alpha = (\alpha_1, \dots, \alpha_l)$ corresponde a un vector de multiplicadores de Lagrange positivos y que se encuentran asociados con las constantes en E.3.

El teorema de Khun-Tucker cumple un rol importante en la teoría de las SVM, pues de acuerdo a este teorema la solución α_i del problema en E.5 satisface las siguientes condiciones:

$$\begin{aligned} \bar{\alpha}(y_i(\bar{w} \cdot z_i + \bar{b}) - 1 + \bar{\xi}_i) &= 0, \quad i = 1, \dots, l \\ (c - \bar{\alpha}_i) \cdot \bar{\xi}_i &= 0, \quad i = 1, \dots, \end{aligned} \quad \text{E.7}$$

de esta igualdad es posible deducir que los únicos valores $\bar{\alpha}_i \neq 0$ en E.7 son aquellos que para las constantes en E.3 son satisfechos con el signo de igualdad.

Se denomina vector de soporte, al punto x_i correspondiente con $\bar{\alpha}_i > 0$, existen dos tipos de vectores de soporte en un caso no separable, por un lado, en el caso en que $0 < \bar{\alpha}_i < C$, el vector de soporte correspondiente x_i , debe satisfacer las igualdades $y_i(\bar{w} \cdot z_i + \bar{b}) = 1$ y $\bar{\xi}_i = 0$, mientras que en el caso $\bar{\alpha}_i = C$, el vector de soporte x_i correspondiente no satisface E.2, y $\bar{\xi}_i \neq 0$.

Se refiere a estos vectores de soporte como errores, el punto x_i correspondiente con $\alpha_i = 0$ es clasificado correctamente y esta alejado del margen de decisión, tal como se muestra en la Figura 39

Muestra parámetros $\xi(i,j)$.

Para la construcción del hiperplano óptimo $\bar{w} \cdot z + \bar{b}$, se utiliza la siguiente ecuación:

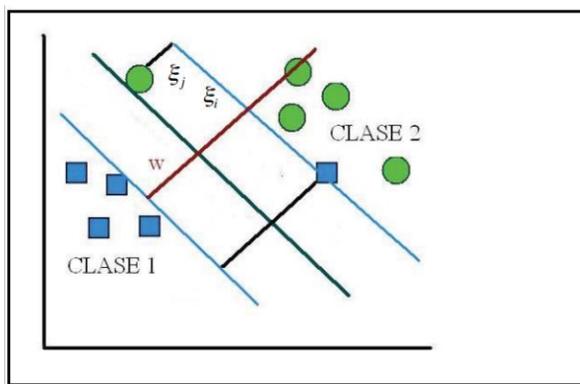
$$\bar{w} = \sum_{i=1}^l \bar{\alpha}_i y_i z_i \text{E.8}$$

y el escalar b puede determinarse de las condiciones del teorema de Kuhn-Tucker, la función generalizada de decisión de E.2 y E.8 es tal que:

$$f(x) = \text{sign}(w \cdot z + b) = \text{sign}\left(\sum_{i=1}^l \alpha_i y_i z_i \cdot z + b\right) \text{ E.9}$$

Figura 39

Muestra parámetros $\zeta(i,j)$



Nota. Datos tomados de (Cuevas Alfaro, 2010).

5.1.8.1.4 Clasificador de máxima verosimilitud. El clasificador de máxima verosimilitud es el algoritmo más utilizado en la percepción remota, debido a su sencilla interpretación de resultados y aplicación; así mismo, se considera uno de los estimadores más acertados y eficientes, esto siempre que los datos sigan una función de distribución normal. (García Mora & Mas, 2008)

El método de máxima verosimilitud está determinado por la aceptación de los datos de entrenamiento para cada clase en cada banda como una distribución normal o Gaussiana. Esto

quiere decir que los datos con histogramas bimodales o trimodales en una única banda no son ideales.

“Utiliza como estadísticas el vector de media (M) y la matriz de covarianza (V) de cada clase para cada banda. La regla que se aplica para un vector desconocido x es: se decide por la clase c , si y sólo si: $P_c \geq P_i$ con $i = 1, 2, \dots, m$ (clases posibles), donde P_c es la probabilidad de que x pertenezca a la clase c .” (Sanz, 2022)

Esto quiere decir que se debe realizar el cálculo del valor de P_c para cada clase con la intención de clasificar un vector de medidas x de un píxel o un objeto desconocido. Luego de realizar este procedimiento, se asigna el objeto a la clase que tenga una mayor probabilidad. Si en determinado caso todas las probabilidades se encuentran por debajo del umbral establecido, el objeto se asigna a la clase desconocida. (Sanz, 2022)

En máxima verosimilitud, se asume que todas las clases tienen la misma probabilidad de suceder, es por esto que dentro de la ecuación todas tienen el mismo valor, siendo: $P(w_j)$ con $j = 1, \dots, m$. A diferencia del clasificador de Bayes, este método de clasificación, no realiza esta suposición, por lo que se aplican dos factores de peso a la estimación de la probabilidad. En primera instancia, el analista determina dicha probabilidad, es decir, una verosimilitud anticipada de ocurrencia para cada clase en el escenario a clasificar. En segunda instancia, existe un peso que va asociado al riesgo para cada clase de realizar mal el procedimiento de clasificación. Al concatenar estos factores, se reduce la probabilidad de ejecutar una mala clasificación,

concluyendo que el proceso de clasificación da como resultado una clasificación teóricamente más optimizada. (Sanz, 2022)

Ambos clasificadores, tanto el de Máxima verosimilitud como el de Bayes, necesitan de un procesamiento computacional demandante, viéndose incrementado cuando se cuentan con una cantidad excesiva de canales de espectro o muchos canales a distinguir.

Algunas de las aplicaciones de este clasificador es el censado remoto para el estudio de los mares congelados en las regiones polares, lo que permite entender el clima y los procesos geofísicos que afectan a los cambios climáticos.

5.1.8.2 Estimación no paramétrica. Estos métodos realizan la clasificación sin contemplar el número de clases o la forma de las distribuciones de probabilidad. (Gómez Gil & Sucar Succar, 2019)

5.1.8.2.1 Boosting. “El boosting es un método utilizado en el machine learning para reducir los errores en el análisis predictivo de datos”. (Amazon, 2022) El entrenamiento del programa automatizado se realiza a través de un modelo de aprendizaje con datos previamente etiquetados que permiten realizar predicciones sobre datos que aún no han sido etiquetados. Si el proceso de aprendizaje se realiza bajo un único modelo se pueden presentar errores de predicción de acuerdo al nivel de confianza del conjunto de datos del entrenamiento, es decir, si un modelo fue entrenado únicamente con un conjunto de datos específicos que forman parte de una clase general de datos con diversas características, existe la posibilidad que el resultado no garantice la identificación de todos los datos que pertenezcan a la misma clase. Por esta razón, el boosting se encarga de realizar un entrenamiento secuencial de múltiples modelos que fortalezcan la precisión de identificación general del sistema.

La base para realizar clasificaciones de tipo no lineal está determinada por la construcción de estructuras cada vez más sofisticadas que van consolidándose gracias a la inclusión de un parámetro tras otro. “Es decir, primero se escoge la característica que mejor discrimina entre las clases. Enseguida, se entra a un ciclo en donde se considera este parámetro anterior fijo y se agrega uno nuevo siempre y cuando mejore la función objetivo”. (Salas, 2018)

Suponiendo que se tiene la función de distribución de probabilidades:

$$p(\mathbf{w}_i|\mathbf{x}_i) = \text{Bern}(\text{sig}(a_i))$$

Siendo $a_i = \phi^T \mathbf{z}_i = \phi^T \mathbf{f}(\mathbf{x}_i)$ el término de activación. De igual forma, consideremos la notación expandida de a_i para un conjunto de parámetros ξ_k como:

$$a_i = \phi_0 + \sum_{k=1}^k \phi_k \mathbf{f}(\mathbf{x}_i, \xi_k)$$

Algunas funciones que comúnmente se utilizan incluyen:

- $\mathbf{f}(\mathbf{x}, \xi) = \arctan(\alpha^T \mathbf{x})$, para parámetros $\alpha = \xi$, ó
- $\mathbf{f}(\mathbf{x}, \xi) = \exp\left(-\frac{1}{2(\mathbf{x}-\alpha)}\right)^T (\mathbf{x} - \alpha) / \lambda^T$.

El aprendizaje incremental procede de la siguiente manera:

- Primero se define un término de activación con una variable, tal como:

$$a_i = \phi_0 + \phi_1 \mathbf{f}(\mathbf{x}_i, \xi_1)$$

con la variable ϕ_1 que mejor ayuda a distinguir las clases. Los parámetros ϕ_0, ϕ_1 y ξ_1 pueden ser aprendidos mediante máxima verosimilitud.

- Enseguida se agrega un segundo conjunto de variables ϕ_2 y ξ_2 , de tal forma que se crea el término de activación:

$$a_i = \phi_0 + \phi_1 \mathbf{f}(\mathbf{x}_i, \xi_1) + \phi_2 \mathbf{f}(\mathbf{x}_i, \xi_2)$$

dejando fijo tanto ϕ_1 como ξ_1 . Nuevamente los términos ϕ_0 , ϕ_2 y ξ_2 se aprenden mediante máxima verosimilitud.

- Esto se sigue mientras las variables que se incluyen agreguen valor al clasificador.

En particular para el término k -ésimo se tiene la función de activación:

$$a_i = \phi_0 + \sum_{k=1}^k \phi_k \mathbf{f}(\mathbf{x}_i, \xi_k)$$

donde se aprenden los términos ϕ_0 , ϕ_k y ξ_k y los demás se mantienen fijos.

“Un caso particular de esta estrategia incremental se da al usar la función escalón, Escalón, toma el lugar de la función \mathbf{f} . La función Escalón (x) se define como uno para valores de x mayores o iguales a cero y cero para todos los demás”. (Salas, 2018)

A pesar que dicha función posee un nivel de complejidad ínfimo, permite establecer la base de distinción entre clases y proyecta que a largo plazo mejore su desempeño de acuerdo con decisiones específicas.

La idea fundamental es que los clasificadores débiles sean robustos en determinar lo que no se requiere clasificar, estableciendo un margen de falsos positivos que al ser concatenados con el resultado de muchos más clasificadores débiles con igual robustez, permitan la construcción de un clasificador fuerte. Es así como de estas premisas nace la estrategia de clasificación conocida como: *boosting*.

Los parámetros se obtienen de forma iterativa mediante el método de Newton:

$$\phi^t = \phi^{t-1} + \alpha(\partial^2 L / \partial \phi^2)^{-1} \partial L / \partial \phi$$

lo cual requiere la evaluación del gradiente:

$$\frac{\partial L(\phi | \mathbf{X}, \mathbf{w})}{\partial \phi} = - \sum_{i=1}^I \left(\frac{1}{1 + \exp(-x_i^T \phi)} - \phi_i \right) x_i$$

y el Hessiano:

$$\frac{\partial^2 L(\phi | \mathbf{X}, \mathbf{w})}{\partial \phi^2} = - \sum_{i=1}^I \left(\frac{1}{1 + \exp(-x_i^T \phi)} \right) \left(\frac{\exp(-x_i^T \phi)}{1 + \exp(-x_i^T \phi)} \right) x_i x_i^T$$

Siguiendo las dos últimas expresiones se permite considerar que la importancia de cada una depende del resultado de la predicción. “Los clasificadores posteriores se enfocan en las partes más difíciles del problema, aquellos que no fueron bien clasificados por otros antes.”

(Salas, 2018)

5.1.8.2.2 Árboles de decisión. Esta técnica se enfoca en desarrollar métodos para el aprendizaje de modelos comprensibles, que se basan en sistemas de reglas. Este método es uno de los más fáciles de entender y utilizar ya que está basado en una estructura jerárquica, compuesta por nodos internos que suelen representar evaluaciones sobre ciertos atributos o características del problema que se desea resolver. En este modelo la decisión final se determina verificando el cumplimiento de condiciones desde la raíz (nodo padre) hasta alguna de las hojas (nodos hijos) por medio de estimaciones. La rama que logra conectar a un par de nodos representa uno de los posibles valores para el atributo del nodo padre, de esta forma simula el proceso de razonamiento normal del ser humano. (Carranza Astrada, 2015)

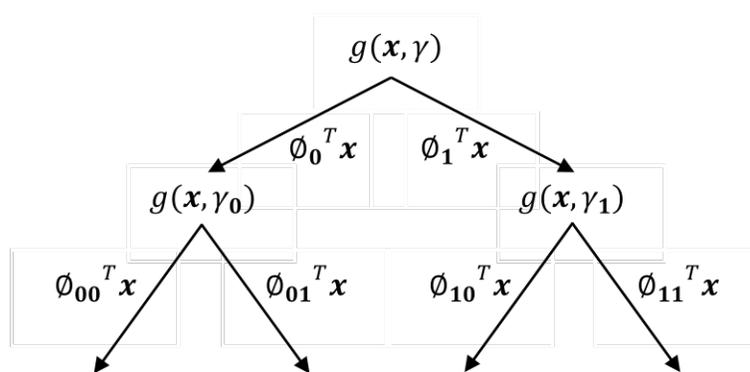
Los árboles de decisión son una herramienta de clasificación sin métrica que permiten analizar decisiones secuenciales. Se basan en el uso de resultados, probabilidades asociadas y usualmente se representan con notación de grafos para su visualización. La idea principal es una vez obtenida un conjunto de datos etiquetados (X_n, Y_n) con $1 \leq n \leq N$, donde $X_n \in R_n$ y $Y_n \in C$, $C = 1 \dots c$, se logre construir un modelo de clasificación en el cual cada nuevo dato X_n pueda ser asociado a una clase C . (Riverón & Marroig, 2011)

Existen diversos tipos de árboles de decisión, entre ellos se destacan los árboles binarios y los árboles *m-arios*. Los primeros, se caracterizan por tener exactamente 2 hijos o ninguno (las hojas); su principal ventaja es que su construcción es más fácil de automatizar utilizando algoritmos computacionales. Por otra parte, en los árboles *m-arios* los nodos generalmente tienen

una mayor cantidad de hijos, de igual forma, es posible representar cualquier tipo de árbol como uno binario. (Riverón & Marroig, 2011)

Figura 40

Árbol de decisión. El espacio es dividido por las superficies de decisión. La superficie que se escoge es una combinación lineal determinada por las funciones de activación



Nota. Datos tomados de (Carranza Astrada, 2015).

Los árboles de decisión suelen ser clasificadores intuitivos, el proceso de decisión en ellos puede ser generado a partir de estimaciones simples, además, se pueden crear a partir de características cuantitativas y cualitativas, lo cual permite su aplicación a problemas discretos y continuos. Sin embargo, son algoritmos inestables, ya que cualquier variación en el conjunto de datos de entrenamiento puede alterar el resultado del clasificador. Esta afirmación se puede comprobar claramente al observar los resultados que se obtienen cuando se vuelve a ejecutar el procedimiento de partición del conjunto de datos y construcción del árbol. (Riverón & Marroig, 2011)

5.1.8.2.3 Bosques aleatorios. Los bosques aleatorios forman parte de una generalización de la metodología Baggin, planteada por Breiman en 1996, donde se propone una arquitectura constituida por múltiples árboles basados en el algoritmo CART, el cual combina el resultado de las predicciones o clasificaciones entregadas por cada árbol de manera independiente. El objetivo principal de realizar esta combinación de algoritmos es lograr reducir el nivel de varianza de la respuesta independiente de cada árbol frente a datos de prueba, con el propósito de optimizar el trabajo del método. (Cardona Alzate, Ospina Arango, & Correa Morales, 2019)

Los bosques aleatorios forman parte de un conjunto de métodos de ensamble encargados de potenciar las capacidades de métodos simples, con el fin de fortalecer la precisión en el resultado de las predicciones. (Cardona Alzate, Ospina Arango, & Correa Morales, 2019)

En los bosques aleatorios se extrae un valor global de predicción o clasificación para la respuesta. Para el caso de regresión este valor se obtiene del promedio de las predicciones generadas por cada árbol del bosque; sin embargo, en el caso de la clasificación se obtiene al elegir la clase con mayor cantidad de votos del conjunto de árboles del bosque, es decir la respuesta más frecuente. (Cardona Alzate, Ospina Arango, & Correa Morales, 2019)

Para construir cada uno de los árboles que conforman el bosque, se realizan cortes binarios, para lo que es necesario definir una o varias variables predictoras p , seleccionadas al azar $m < p$ predictores que van a ser los candidatos para el corte. (Fernandez Casal, 2022)

De igual forma, se establece m como un hiperparámetro de los árboles aleatorios, es decir, como un conjunto de valores de configuración, determinados durante el proceso de entrenamiento.

Es considerable tener en cuenta que la configuración para problemas de clasificación, se puede establecer como: $m = \sqrt{p}$ y para problemas de regresión como: $m = \frac{p}{3}$. El número de árboles por bosque podría determinarse como un hiperparámetro, aunque, habitualmente se emplea más como un problema de convergencia. (Fernandez Casal, 2022)

A pesar de que el método de Bosques Aleatorios, forma parte de una generalización de la metodología Baggin, son computacionalmente más eficientes, requieren más árboles por bosque, pero el proceso de construcción de cada bosque es más efectivo debido a que en cada corte se evalúan unos pocos predictores. (Fernandez Casal, 2022)

5.1.8.3 Métodos estocásticos. Usan funciones aleatorias para diseñar las funciones de clasificación. (Gómez Gil & Sucar Succar, 2019)

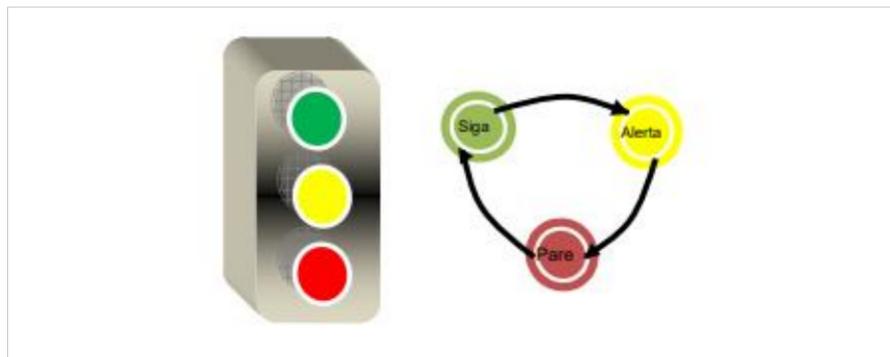
5.1.8.3.1 Modelos de Markov. Es una técnica de clasificación estocástica, que modela el comportamiento de procesos aleatorios en el transcurso del tiempo, utilizando una familia de variables aleatorias parametrizadas y estimadas a través de la identificación de su comportamiento secuencial con un margen de probabilidad previa.

Normalmente un proceso aleatorio se caracteriza por presentar una serie de eventos que no siguen una regla preestablecida, por lo tanto, no es fácil definir con certeza una secuencia de eventos previamente (a priori) de forma segura antes de que estos ocurran. Por el contrario, algo que se puede definir con certeza es la secuencia previa de eventos de operación de un semáforo (Verde-amarillo-rojo-verde), antes que termine el ciclo completo. En la Figura 41

Proceso determinista: Iluminación del semáforo, se observa este proceso representado como una máquina de estados. (Ospina Mateus, 2008)

Figura 41

Proceso determinista: Iluminación del semáforo



Nota. Datos tomados de (Ospina Mateus, 2008).

Este sistema trabaja bajo reglas deterministas que le indican la secuencia de eventos exacta a seguir para la que la siguiente operación tenga sentido. Sin embargo, en las cadenas de markov ocurre lo contrario se conoce cuáles son los eventos posibles de un proceso, pero no se conoce con absoluta certeza cómo se comportan a través del tiempo. Además, muchas de las aplicaciones científicas hacen uso de mecanismos más potentes debido a que la naturaleza de algunos procesos contiene estados que no son visibles, medibles o accesibles, inhabilitando la

posibilidad de utilizar cadenas de markov. Un método alternativo, es el de seleccionar variables accesibles y medibles que estén probabilísticamente relacionados a los estados ocultos del sistema. (Ospina Mateus, 2008)

5.1.8.3.2 Modelos ocultos de Markov. Los modelos ocultos de Markov, surgieron a partir de la necesidad de caracterizar procesos estocásticos, en los cuales las observaciones obtenidas estaban incompletas. La única forma de no prescindir de las observaciones como partes fundamentales de los sistemas de predicción, era adoptar un nuevo pensamiento en donde se fundamenta que las observaciones siguen un patrón de comportamiento oculto y por tanto la ausencia de algunas observaciones ya no era tan relevante. Este proceso se desarrolla en la secuencia de estados ocultos en los cuales, dependiendo de las características del proceso a modelar, tengan un comportamiento o secuencia posiblemente definida. De aquí surgió la idea de los modelos ocultos de Markov como un sistema doblemente estocástico, es decir, que presentaba aleatoriedad tanto en las observaciones, como en la secuencia de estados ocultos que las gobiernan. Lo más interesante aún es que los dos procesos estocásticos deben ser definidos en la etapa de entrenamiento solamente a través de las observaciones. (Ospina Mateus, 2008)

Como se mencionó, su estructura consta de dos capas estocásticas. La primera capa consiste en un proceso de estados ocultos secuenciales en el cual no son directamente observables y que obedecen a las reglas de transición de los modelos de markov que se mencionaron anteriormente. La segunda capa es un proceso de emisión de símbolos de observación, en el cual es fácilmente observable y tienen cierta función de distribución de

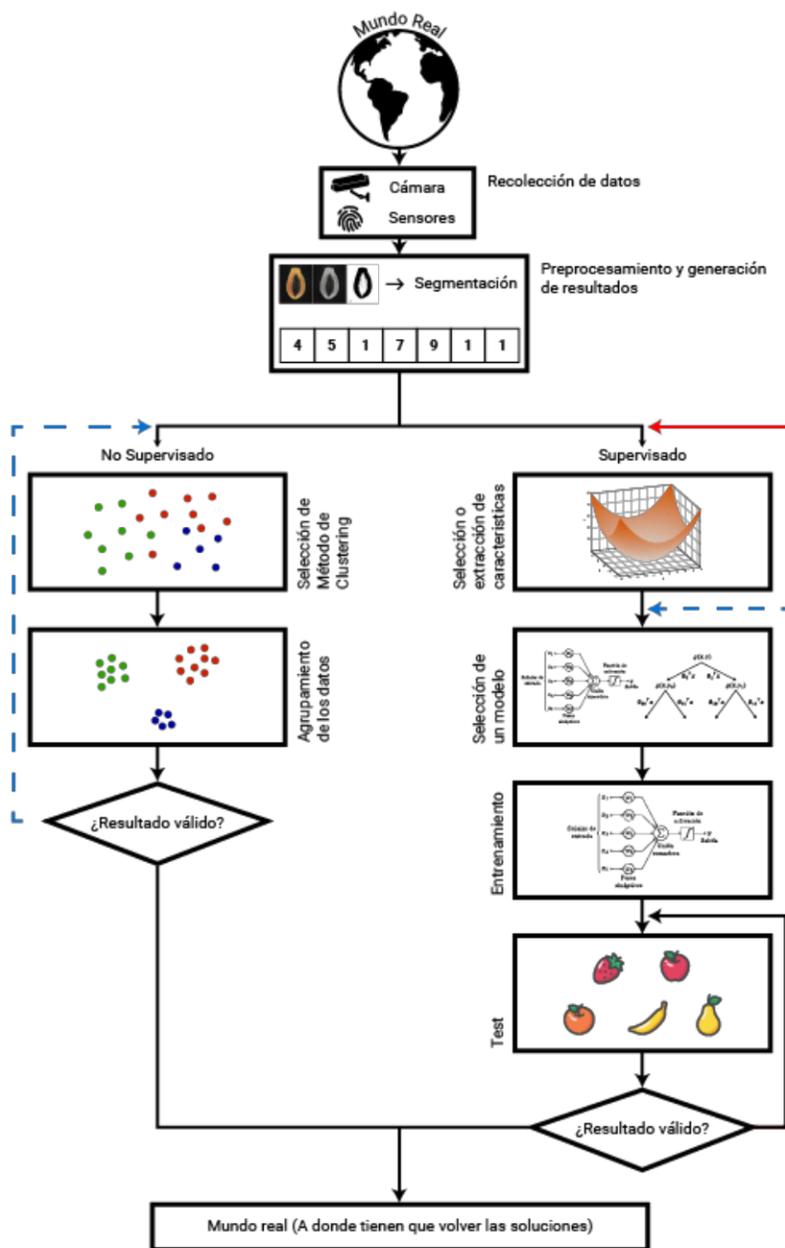
probabilidad dependiendo del estado oculto en el que se encuentra actualmente. (Ospina Mateus, 2008)

Este tipo de modelo restringe frecuentemente la aplicabilidad a diferentes problemas de interés del mundo real. La mayoría de los procesos a modelar, experimentan cambios de estado que sugieren algún significado (simbólico o no simbólico) relevante e indispensable a los científicos que se dedican a analizar los impactos que representan estos cambios. Sin embargo, para muchas aplicaciones, estos estados no son fácilmente identificables a primera vista (no observable), lo que conlleva a la inspección de variables externas observables (cuantitativas o cualitativas) relacionadas a los estados ocultos del proceso y a partir de estas inferir la secuencia de estados por los cuales se ha desarrollado. Es por esto que un conjunto de variables observables del proceso con valores o características definidas en un instante dado, puede dar información del posible estado en que se encuentra dicho proceso, teniendo en cuenta además que las observaciones pueden ser ruidosas, imprecisas, erróneas o en el peor de los casos estar incompletas.

5.1.8.4 Métodos basados en aprendizaje supervisado / no supervisado. Utilizan modelos de aprendizaje de máquina para definir las funciones. Este aprendizaje se lleva a cabo a través de ejemplos de los patrones a clasificar, los cuales se conocen como el conjunto de entrenamiento. Cuando se conoce previamente el valor de clase de los objetos en el conjunto de entrenamiento, se dice que el aprendizaje es supervisado; en caso contrario el aprendizaje se denomina “no supervisado”. (Gómez Gil & Sucar Succar, 2019)

Figura 42

Esquema del ciclo de diseño de un clasificador



Nota. Datos tomados de (Santa Catarina, Marmolejo Cortes, Fiorio Vettorazzi, & Gonzaga Pereira, 2018)

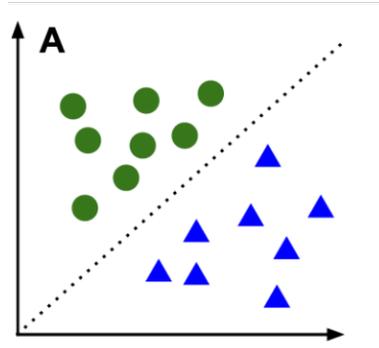
5.1.8.4.1 Supervisados. Los algoritmos de clasificación basados en aprendizaje supervisado más comunes, son:

- Modelos lineales
- Método de clasificación de paralelepípedo (PIPED)
- Regresión lineal
- Regresión logística

En el **modelo lineal**, los clasificadores lineales reconocen patrones a través de cálculos matemáticos. Estos clasificadores funcionan bien para problemas prácticos como la clasificación de documentos y, en general, para problemas con muchas variables (características), alcanzando niveles de precisión comparables a los de los clasificadores no lineales con menos tiempo de entrenamiento y uso. (Hmong, 2022)

Dentro de los más comunes encontramos:

- **Modelo lineal basado en regionalización**, el clasificador basado en regionalización particiona el espacio de características en (R) regiones de decisión denominadas clases. Estas clases generalmente deben cubrir todo el espacio R y ser disjuntas (no tener ningún elemento en común) tal como lo muestra la Figura 43 Clasificación lineal, para lograr la asignación de una posible clase única. (Sanz, 2022)

Figura 43*Clasificación lineal*

Nota. Datos tomados de (Palomera & Figueroa, 2015).

El borde de cada clase es un límite de decisión. En base a lo anterior, la clasificación de un vector desconocido X , resulta un proceso bastante sencillo, ya que se determina la región de decisión (en R) a la que pertenece X y se le asigna X a esa clase. Sin embargo, a pesar de que este modelo es directo, la clasificación de las regiones de decisión son un desafío, ya que las funciones discriminantes en el caso de n características, pasan a ser hiperplanos.

- **Modelo lineal basado en distancia**, en este modelo se considera que se tienen M clases y que cada una se ve representada por un vector prototipo, tal como lo explica la siguiente ecuación: (Sanz, 2022)

$$m_j = \frac{1}{N_j} \sum_{x \in W_j} x \quad j = 1 \dots$$

N_j : Es el número de vectores patrones para la clase W_j . La sumatoria se toma sobre estos vectores. (Sanz, 2022)

Una forma de determinar a qué clase pertenece el patrón desconocido X , es asignarlo a la clase más cercana a su prototipo, a través del cálculo de la similitud con cada clase computando su distancia al vector prototipo de la clase. Para encontrar la proximidad o distancia de la clase a clasificar se suele usar el modelo de distancia Euclidiana, ya que al usar este método se reducen las distancias del problema a solucionar. (Sanz, 2022)

$$d_j = \|x - m_j\| \quad j = 1 \dots M$$

$$\|a\| = (a^T a)^{\frac{1}{2}} \text{ es la norma euclidiana.}$$

Entonces, asignamos x a la clase w_i , si $d_i(x)$ es la menor distancia encontrada. Esto es, la menor distancia implica la mejor concordancia o “matching” en esta formulación. No es difícil demostrar que esto es equivalente a evaluar las funciones:

$$d_j = x^T m_j - \frac{1}{2} m_j^T m_j \quad j = 1 \dots M$$

Se asigna x , a la clase w_i , si $d_i(x)$ lleva al valor numérico más grande. El límite de decisión entre la clase w_i y w_j para un clasificador de mínima distancia es:

$$d_{ij} = d_i(x) - d_j(x) = 0$$

En la clasificación de datos censados remotamente, es necesario que el usuario provea los vectores medios para cada clase en cada banda. (Sanz, 2022)

La principal ventaja de este clasificador es su simplicidad y eficiencia computacionalmente. Sin embargo, uno de sus problemas es su insensibilidad a los diferentes niveles de varianza en los datos de respuesta espectral. Por lo que, su uso, no es apto para aplicaciones que poseen un alto grado de varianza y en las que las clases están cerca unas de otras en el espacio de medidas.

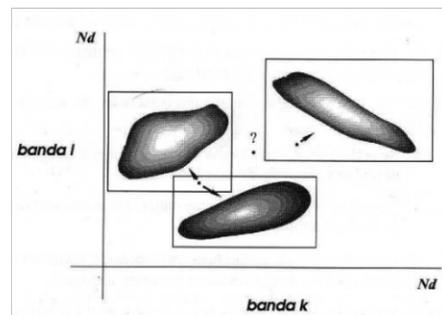
Por otra parte, el *Método de clasificación de paralelepípedo (PIPED)*, es el más rápido de los clasificadores supervisados y además puede introducir sensibilidad a la varianza considerando un rango de valores en cada conjunto de entrenamiento. El rango puede estar dado, por ejemplo, por el valor más bajo y más alto en cada banda. (INRED, 2007)

La clasificación de PIPED se basa en una serie de umbrales bajos y altos de reflectividad para cada categoría en cada banda, determinadas por la signatura espectral. Para asignar un píxel

a una clase específica, este debe poseer valores de reflectividad dentro del rango de la categoría en cuestión en cada banda considerada. (INRED, 2007)

Figura 44

Clasificador de paralelepípedos



Nota. Datos tomados de (INRED, 2007).

En este método se usan para clasificar los datos de entrenamiento una cantidad de n bandas espectrales. Se tiene un vector de n -dimensional. (Sanz, 2022)

$$M_c = (U_{ck}, U_{c2}, \dots, U_{cn})$$

Donde U_{ck} es el valor medio de los datos de entrenamiento obtenidos para la clase c en la banda k , entre m clases posibles. Se llama S_{ck} a la desviación estándar de los datos de entrenamiento de la clase c en la banda k . (Sanz, 2022)

Usando un umbral basado en la desviación estándar y siendo Bv_{ijk} el valor del pixel de la banda k en las coordenadas espaciales (i, j) , el algoritmo del paralelepípedo decide que Bv_{ijk} está en la clase c , sí y sólo si: (Sanz, 2022)

$$(U_{ck} - S_{ck}) \leq Bv_{ijk} \leq (U_{ck} + S_{ck})$$

$c = 1, 2, \dots, m$. Número de clases.

$k = 1, 2, 3 \dots, n$. Número de bandas.

Si definimos los límites (inferior y superior) de decisión como:

$$L_{ck} = U_{ck} - S_{ck} \text{ y } H_{ck} = U_{ck} + S_{ck}$$

El algoritmo del paralelepípedo se convierte en:

$$L_{ck} \leq Bv_{ijk} \leq H_{ck}$$

Los límites de decisión forman un paralelepípedo n -dimensional en el espacio de rasgos.

Si el valor de un píxel está por arriba del umbral inferior y por debajo del superior para todas las bandas evaluadas, se asigna el pixel a la clase. Cuando el pixel no satisface este criterio para ninguna de las clases se asigna a la clase “desconocida”. (Sanz, 2022)

El algoritmo de PIPED es eficiente computacionalmente para clasificar datos censados remotamente, sin embargo, algunos paralelepípedos se superponen; esto quiere que un píxel puede satisfacer las condiciones para más de una clase. (Sanz, 2022)

En el algoritmo de *regresión lineal*, “Si disponemos de un conjunto de datos $S = \{(x^k, y^k), k = 1, \dots, N\}$, donde tratamos de predecir un valor numérico continuo, $y^k \in R$, podemos emplear un modelo de regresión lineal.” La idea principal de la regresión líneas es determinar una función h idónea que relacione las variables independientes (descriptores) y la variable dependiente (etiqueta numérica), la función h que se requiere estará determinada por una expresión de combinación lineal de los descriptores con pesos θ , siendo de la forma: (Alegre, Pajares, & De la Escalera, 2016)

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \dots + \theta_m x_m$$

“El ajuste de los parámetros θ_i ha de hacerse con el conjunto de datos disponibles S , de manera que $h_{\theta}(x^k)$ sea lo más próximo a y^k . Para calcular el error existente entre los datos reales y los datos estimados por nuestra función, definimos una función de coste, el error cuadrático medio:” (Alegre, Pajares, & De la Escalera, 2016)

$$J(\theta) = \frac{1}{2N} \sum_{i=1}^N (h_{\theta}(x) - y)^2$$

Donde N se corresponde con el número de datos.

Es necesario seleccionar los parámetros de θ que minimizan la función de coste $J(\theta)$, por esta razón, se pueden inicializar los parámetros de forma aleatoria y posteriormente aplicar un algoritmo de búsqueda como el *algoritmo de descenso de gradiente*, que cambia consecutivamente el valor de θ , haciendo que el valor de $J(\theta)$ sea cada vez menor y permita establecer el valor de los parámetros que posibilitan disminuir la función de coste.

“El algoritmo de descenso por gradiente comienza un con valor arbitrario para los parámetros, que se actualizan de la forma:

$$\theta_j = \theta_j - \alpha \frac{d}{d\theta_j} J(\theta)$$

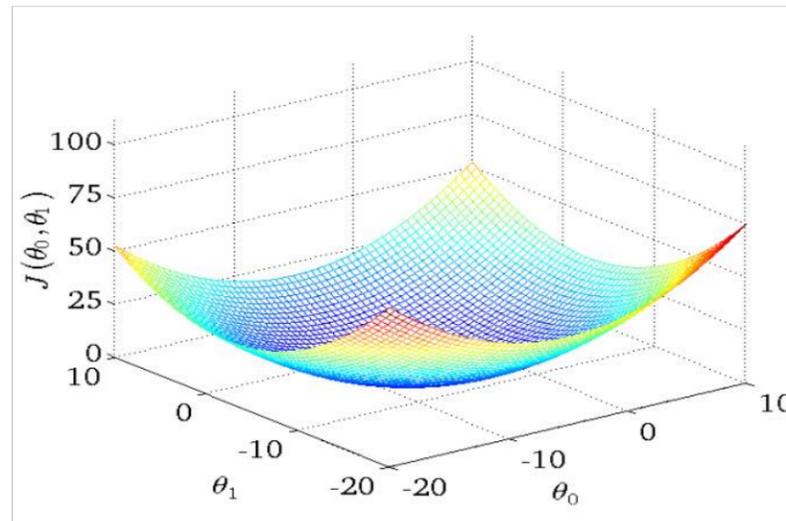
donde θ es la tasa de aprendizaje, que indica cuánto cambian los parámetros y el signo menos indica que nos movemos en la dirección opuesta al gradiente (ya que la dirección del gradiente es la de máximo incremento de la función).” (Alegre, Pajares, & De la Escalera, 2016) Es importante tener en cuenta que, si se determina un valor muy pequeño para α , la función se retrasará en determinar el valor de los parámetros necesarios. En cambio, si se determina un valor muy grande, la función oscilará, por lo que puede que no se lleguen a determinar nunca los valores mínimos esperados.

Ilustraremos este proceso, con un problema de regresión con dos parámetros donde la función de regresión lineal toma la forma $h_{\theta}(x) = \theta_0 + \theta_1 x$.

La Figura 45 muestra un ejemplo de la función de coste $J(\theta)$ donde podemos observar que es una función convexa con un mínimo global.

Figura 45

Representación de la función de coste para un problema de regresión lineal con dos parámetros



Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016)

Particularizando la ecuación anterior para un caso con dos parámetros, nos queda la siguiente ecuación:

$$\frac{d}{d\theta_j} J(\theta_0, \theta_1) = \frac{d}{d\theta_j} \frac{1}{2N} \sum_{i=1}^N (h_{\theta}(x) - y)^2 = \frac{d}{d\theta_j} \frac{1}{2N} \sum_{i=1}^N (\theta_0 + \theta_i x_1 - y)^2$$

Si derivamos ahora con relación a ambos parámetros, tenemos:

Para θ_0 :

$$\frac{d}{d\theta_0} J(\theta_0, \theta_1) = \frac{1}{N} \sum_{i=1}^N (h_{\theta}(x) - y)$$

Para θ_1 :

$$\frac{d}{d\theta_1} J(\theta_0, \theta_1) = \frac{1}{N} \sum_{i=1}^N (h_{\theta}(x) - y) x$$

Por tanto, el algoritmo de descenso de gradiente aplicado al problema de regresión lineal, se resume como:

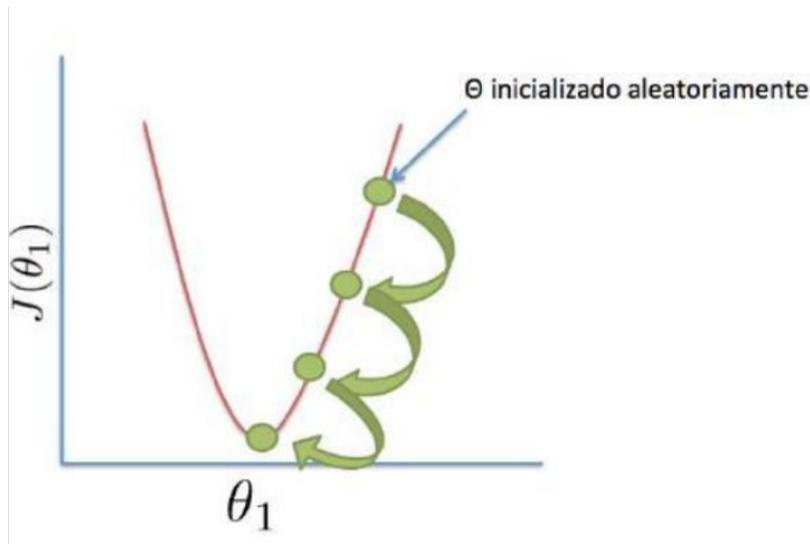
$$\theta_0 = \theta_0 - \alpha \frac{1}{N} \sum_{i=1}^N (h_{\theta}(x) - y)$$

$$\theta_1 = \theta_1 - \alpha \frac{1}{N} \sum_{i=1}^N (h_{\theta}(x) - y) x$$

La Figura 46 muestra cómo sería la actualización del parámetro θ_1 con el algoritmo de descenso de gradiente.

Figura 46

Descenso por gradiente



Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

En el clasificador de *regresión logística*, las técnicas de regresión pueden ser adaptadas de forma sencilla para dar solución a problemas de clasificación supervisada teniendo en cuenta un número discreto de clases. El fundamento de esta idea está dado por la ejecución de la regresión para cada clase, asignando el valor uno (1) para los ejemplos que pertenecen a la clase de interés y cero (0) para el resto.

“En este tipo de clasificación queremos que nuestra etiqueta de salida sea 0 o 1, por tanto, podemos acotar la función de regresión lineal entre esos valores. Para ello, vamos a emplear la función sigmoïdal, que viene dada por:” (Alegre, Pajares, & De la Escalera, 2016)

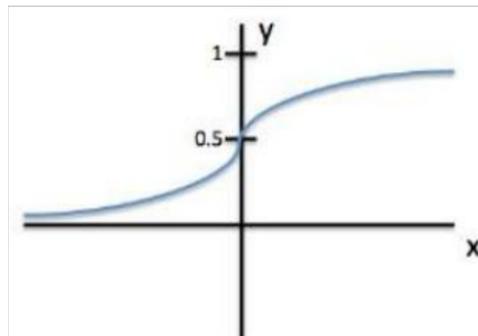
$$g(z) = \frac{1}{1 + e^{-z}}$$

y se representa de la forma mostrada en la Figura 47

Ejemplo de funcionamiento del ajuste de los parámetros:

Figura 47

Ejemplo de funcionamiento del ajuste de los parámetros



Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

De esta manera, nuestra hipótesis quedaría acotada:

$$h_{\theta} = g(\theta_0 + \theta_1 x_1 + \dots + \theta_m x_m)$$

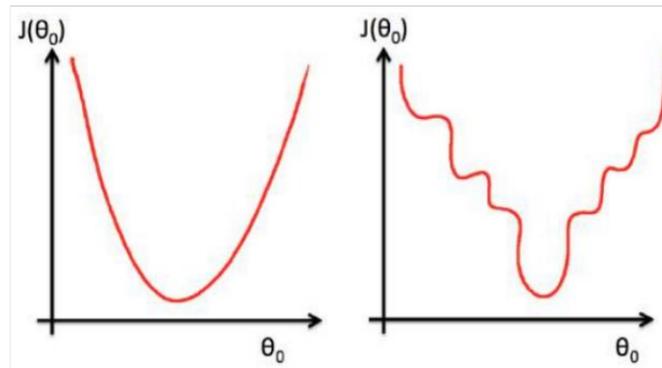
Podemos decir que $h_{\theta}(x^k)$ representa la probabilidad de que la salida sea uno, dado el dato x^k , y parametrizado con θ .

No se puede utilizar la misma función de coste utilizada para los problemas de regresión lineal, debido a que la función fue una acotación de resultados, por lo que se generaría una representación escalonada, similar a la de la gráfica mostrada en la Figura 48

Diferencia entre la función coste de regresión lineal (izquierda) y regresión logística (derecha).

Figura 48

Diferencia entre la función coste de regresión lineal (izquierda) y regresión logística (derecha)



Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

“Si utilizáramos la misma función coste para ambos métodos, en el caso de la regresión logística, es probable que no consiguiéramos llegar a encontrar el mínimo global, por la existencia de muchos mínimos locales.” (Alegre, Pajares, & De la Escalera, 2016)

La función coste que emplearemos para regresión logística es la siguiente:

$$J(\theta) = -\frac{1}{N} \left[\sum_{i=1}^N d \log h_{\theta}(x) + (1 - d) \log(1 - h_{\theta}(x)) \right]$$

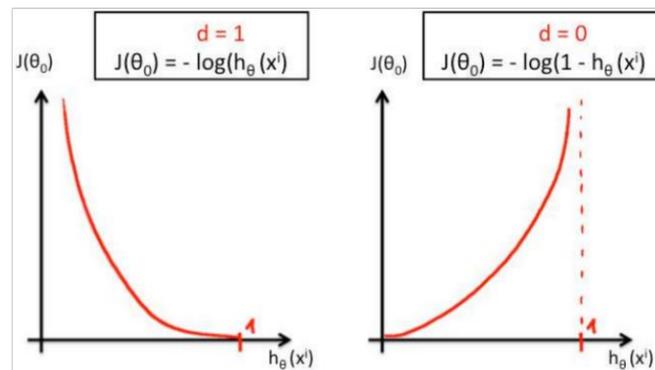
“La Figura 49

Función coste de la regresión logística muestra la contribución a la función de coste para: un ejemplo x^i que pertenece a la clase 1 (izquierda) y un ejemplo que pertenece a la clase 0 (derecha). Cuando la etiqueta es $d = 1$, el segundo sumando de la ecuación anterior queda

anulado y si nuestra salida de la función se aproxima a uno (que es la etiqueta real), el error tiende a cero. En cambio, si se aproxima a cero, el error tiende a infinito”

Figura 49

Función coste de la regresión logística



Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Ajustado el modelo de clasificación, retornará para los datos de testeo, un valor en el intervalo de cero a uno que indicará la probabilidad de pertenencia a la clase uno. Finalmente se aplicará un umbral que determinará la decisión final del clasificador y su pertenencia a la clase.

5.1.8.4.2 No supervisados. Dentro de los algoritmos supervisados encontramos el clasificador *K-means*, es un algoritmo popular de clasificación no supervisada que se encarga de organizar los datos registrados en grupos o también denominados: clusters. La idea fundamental del algoritmo es que los grupos cuenten con características próximas y similares entre sí, pero distintivas de los demás grupos. Para poder agrupar el conjunto de datos, se procede a dividir en un sin fin de observaciones cierta cantidad de grupos, haciendo que cada observación pertenezca al grupo más cercano. (Alegre, Pajares, & De la Escalera, 2016)

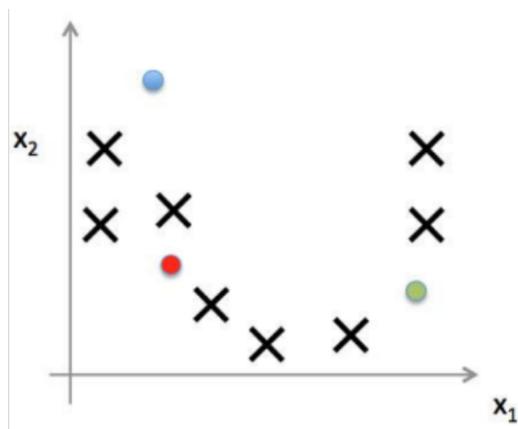
De acuerdo con esta premisa, el espacio de datos se distribuye en celdas de Voronoi, determinadas por el nodo de mayor representación respecto al valor de cercanía del conjunto de datos del espacio geométrico de los demás nodos. En primera instancia el usuario determina el número de grupos (k) en el que se van a agrupar los datos.

Una forma práctica de entender el algoritmo es a través del siguiente ejemplo, donde se describen las actividades de cada paso:

Paso 1: Se determina el número de grupos, inicializando aleatoriamente los k centros. Dentro del ejemplo se determina que k equivale a 3 ($k = 3$), representados por los colores rojo, verde y azul.

Figura 50

Paso 1: Inicialización aleatoria de los centros

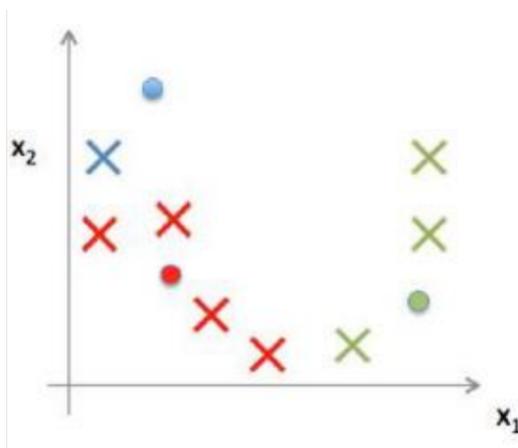


Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Paso 2: Se realiza la asignación de datos a los centros en función de la distancia del dato con el centro. Como se aprecia en la figura, el color del dato X está representado por el centro al que fue asignada en función de su distancia más próxima.

Figura 51

Paso 2: Asignación de puntos a centros en función de las distancias

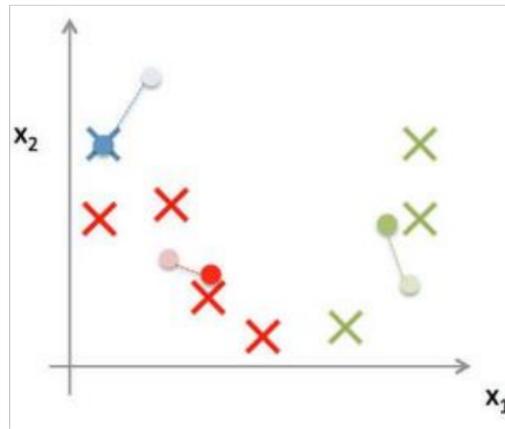


Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Paso 3: Se reubica el centro de acuerdo con la media del conjunto de datos asignados al grupo.

Figura 52

Paso 3: El centro se reubica pasando a ser el centroide de todos los puntos que tenía asignados



Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Paso 4: Se repiten los pasos 2 y 3 donde se asignan los puntos y se reubican los centros, hasta que, dentro de dos iteraciones consecutivas, la posición de los centros no se vea alterada.

Figura 53

Algoritmo K-means.

Entrada: k , conjunto de N datos sin etiquetar: $\{x^1, x^2, \dots, x^n\}$

1. Inicializar aleatoriamente los k centros dentro del espacio representado por los objetos.
2. Asignar cada muestra al centro más cercano.
3. Cuando todas las muestras hayan sido asignadas al centro correspondiente, recalcular las posiciones de los centros de manera que se conviertan en el centroide de las muestras que tienen asignadas a dichos centros.
4. Repetir los pasos 2 y 3 hasta que los centros ya no se muevan. Esto produce una separación de los datos en k grupos de manera que muestras más parecidas entre si pertenezcan al mismo grupo.

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Ejemplo: “Se quiere dividir un conjunto de individuos según su peso y altura en dos grupos para intentar determinar el sexo de cada uno de ellos (siendo la variable x_1 el peso y x_2 la altura del individuo). Utilizar KMeans con $k = 2$ para determinar los clusters más apropiados para el siguiente conjunto de datos:” (Alegre, Pajares, & De la Escalera, 2016)

Tabla 6

Datos iniciales - Parte I

	1	2	3	4	5	6	7	8	9	10	11	12	13
x_1	40	44	55	59	55	45	45	65	68	66	65	111	80
x_2	150	148	154	162	165	148	163	160	171	179	170	195	180

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Tabla 7

Datos iniciales - Parte II

	14	15	16	17	18	19	20	21	22	23	24	25
x_1	74	69	74	87	79	95	87	63	60	76	73	70
x_2	172	179	181	173	168	180	192	165	158	181	177	155

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Los dos centroides se han inicializado aleatoriamente a los siguientes valores:

Tabla 8

Inicialización aleatoria de centroides.

	x_1	x_2
$C^01 =$	40	150
$C^02 =$	80	180

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Solución:

Primera iteración

El primer paso consiste en calcular la distancia de cada uno de los elementos de nuestro conjunto de datos a los clusters para determinar a cuál de ellos pertenecen (al más cercano):

Tabla 9

Iteración I.

Centroide más cercano	Vector de datos		Distancia euclídea a C1	Distancia euclídea a C2	
	x_1	x_2			
1	1	40	150	0.0	50.0
1	2	44	148	4.5	48.2
1	3	55	154	15.5	36.1
1	4	59	162	22.5	27.7
1	5	55	165	21.2	29.2
1	6	45	148	5.4	47.4
1	7	45	163	13.9	38.9
2	8	65	160	26.9	25.0
2	9	68	171	35.0	15.0
2	10	66	179	38.9	14.0
2	11	65	170	32.0	18.0
2	12	111	195	84.1	34.4
2	13	80	180	50.0	0.0
2	14	74	172	40.5	10.0
2	15	69	179	41.0	11.0
2	16	74	181	46.0	6.1
2	17	87	173	52.3	9.9
2	18	79	168	43.0	12.0
2	19	95	180	62.6	15.0
2	20	87	192	63.0	13.9
2	21	63	165	27.5	22.7
1	22	60	158	21.5	29.7
2	23	76	181	47.5	4.1
2	24	73	177	42.6	7.6
2	25	70	155	30.4	26.9

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

A continuación, se actualizan los nuevos centroides teniendo en cuenta todos los elementos que han sido asignados a cada cluster. Así pues, los elementos que pertenecen al cluster 1 son:

Tabla 10*Elementos del cluster 1 – Iteración I.*

	1	2	3	4	5	6	7	22
x_1	40	44	55	59	55	45	45	60
x_2	150	148	154	162	165	148	163	158

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

El nuevo centroide será la media de cada coordenada:

Tabla 11*Nuevo centroide 1 – Iteración I.*

	x_1	x_2
C^{11}	50.38	156

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Los elementos que pertenecen al cluster 2 son:

Tabla 12*Elementos del cluster 2 – Iteración I – Parte I*

	8	9	10	11	12	13	14	15
x_1	65	68	66	65	111	80	74	69
x_2	160	171	179	170	195	180	172	179

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Tabla 13*Elementos del cluster 2 – Iteración I – Parte II*

	16	17	18	19	20	21	23	24	25
x_1	74	87	79	95	87	63	76	73	70
x_2	181	173	168	180	192	165	181	177	155

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Por lo que el nuevo centroide C^{12} será:

Tabla 14*Nuevo centroide 2 – Iteración I.*

	x_1	x_2
$C^{12} =$	76.59	175.2

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Como los centroides se han desplazado se continúa con la siguiente iteración:

Segunda iteración

El primer paso es actualizar las distancias de cada elemento a cada centroide para determinar a qué cluster pertenecen:

Tabla 15*Iteración II.*

Centroide más cercano	Vector de datos		Distancia euclídea a C1	Distancia euclídea a C2	
	x_1	x_2			
1	1	40	150	12.0	44.4
1	2	44	148	10.2	42.4
1	3	55	154	5.0	30.2
1	4	59	162	10.5	22.0
1	5	55	165	10.1	23.9
1	6	45	148	9.6	41.7
1	7	45	163	8.8	33.9
1	8	65	160	15.2	19.1
2	9	68	171	23.1	9.5
2	10	66	179	27.8	11.3
2	11	65	170	20.2	12.7
2	12	111	195	72.1	39.7
2	13	80	180	38.1	5.9
2	14	74	172	28.5	4.1
2	15	69	179	29.6	8.5
2	16	74	181	34.4	6.4
2	17	87	173	40.4	10.6
2	18	79	168	31.0	7.6
2	19	95	180	50.7	19.0
2	20	87	192	51.4	19.8
1	21	63	165	15.5	17.0
1	22	60	158	9.8	23.9
2	23	76	181	35.8	5.9
2	24	73	177	30.9	4.0
1	25	70	155	19.7	21.2

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

A continuación, se actualizan los nuevos centroides teniendo en cuenta todos los elementos que han sido asignados a cada cluster. Así pues, los elementos que pertenecen al nuevo cluster 1 son:

Tabla 16

Elementos del cluster 1 – Iteración II.

	1	2	3	4	5	6	7	8	21	22	25
x_1	40	44	55	59	55	45	45	65	63	60	70

x_2	150	148	154	162	165	148	163	160	165	158	155
-------	------------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

El nuevo centroide será la media de cada coordenada:

Tabla 17

Nuevo centroide 1 – Iteración II.

	x_1	x_2
$C^2_1 =$	54.6	157.09

Nota. Datos tomados de

Por lo que el nuevo centroide C^2_2 será:

Tabla 18

Nuevo centroide 2 – Iteración II.

	x_1	x_2
$C^2_2 =$	78.86	178.43

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Como los centroides se han desplazado se continúa con la tercera iteración:

Tercera Iteración

Al igual que en las iteraciones anteriores, el primer paso es actualizar las distancias y los miembros de cada cluster:

Tabla 19

Iteración III.

Centroide más cercano	Vector de datos		Distancia euclídea a C1	Distancia euclídea a C2	
	x_1	x_2			
1	1	40	150	16.3	48.1
1	2	44	148	14.0	46.3
1	3	55	154	3.1	34.1
1	4	59	162	6.6	25.8
1	5	55	165	7.9	27.4
1	6	45	148	13.2	45.5
1	7	45	163	11.3	37.2
1	8	65	160	10.8	23.1
2	9	68	171	19.3	13.2
2	10	66	179	24.7	12.9
2	11	65	170	16.6	16.2
2	12	111	195	67.9	36.2
2	13	80	180	34.2	1.9
2	14	74	172	24.4	8.1
2	15	69	179	26.2	9.9
2	16	74	181	30.8	5.5
2	17	87	173	36.1	9.8
2	18	79	168	26.7	10.4
2	19	95	180	46.4	16.2
2	20	87	192	47.6	15.8
1	21	63	165	11.5	20.8
1	22	60	158	5.4	27.8
2	23	76	181	32.1	3.8
2	24	73	177	27.1	6.0
1	25	70	155	15.5	25.0

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Una vez hecho esto, se calculan los nuevos centroides. Para $C^3 1$:

Tabla 20*Elementos del cluster 1 – Iteración III.*

	1	2	3	4	5	6	7	8	21	22	25
x_1	40	44	55	59	55	45	45	65	63	60	70
x_2	150	148	154	162	165	148	163	160	165	158	155

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

El nuevo centroide será la media de cada coordenada:

Tabla 21*Nuevo centroide 1 – Iteración III*

	x_1	x_2
$C^3 1 =$	54.6	157.09

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Para $C^3 2$:

Tabla 22*Elementos del cluster 3 – Iteración III.*

	9	10	11	12	13	14	15	16	17	18	19	20	23	24
x_1	68	66	65	111	80	74	69	74	87	79	95	87	76	73
x_2	171	179	170	195	180	172	179	181	173	168	180	192	181	177

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Por lo que el nuevo centroide $C^3 2$ será:

Tabla 23

Nuevo centroide 2 – Iteración III.

	x_1	x_2
$C^3_2 =$	78.86	178.43

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

Como los centroides de la iteración 2 y 3 son iguales, el algoritmo se detiene y ya tenemos los clusters creados correctamente dividiendo el conjunto de muestras en función del sexo del individuo:

Tabla 24

Centroides finales.

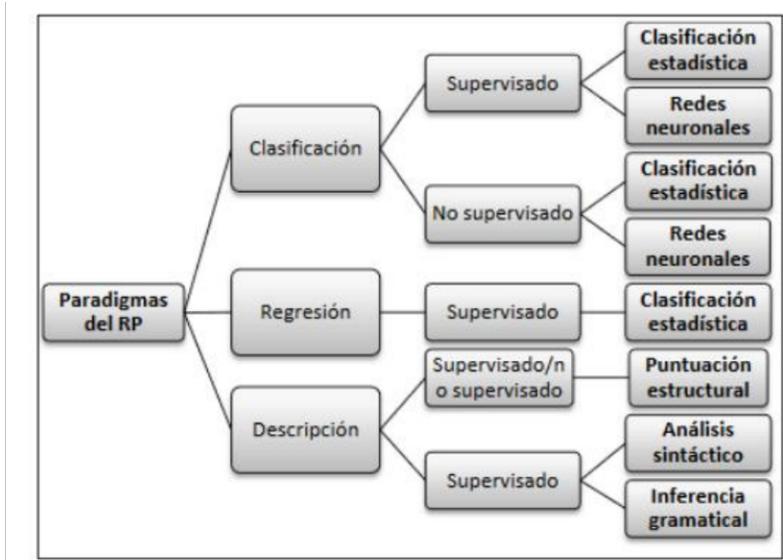
$C^2_1 =$	54.64	157.09	$C^3_1 =$	54.64	157.09
$C^2_2 =$	78.86	178.43	$C^3_2 =$	78.86	178.43

Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

5.1.9 Sistema de reconocimiento de patrones

Figura 54

Paradigmas del Reconocimiento de patrones



Nota. Datos tomados de (Alegre, Pajares, & De la Escalera, 2016).

El reconocimiento de patrones generalmente no requiere esfuerzo alguno para un ser humano, sin embargo, para una máquina, este proceso conlleva resolver problemas que involucran cuestiones de medición, validación, procesamiento e interpretación de la información en el entorno de estudio, haciendo que el reconocimiento de patrones se vuelva un campo de estudio multidisciplinario.

Dado un universo de estudio, el objetivo de un sistema de reconocimiento de patrones consiste en particionar dicho universo en clases "*patrón*" ajenas, de tal manera que el sistema asignará a un elemento x a una y sólo una clase. Es decir, el problema es de clasificación, el sistema reconoce que un elemento pertenece a una clase. (Vazquez, 2022)

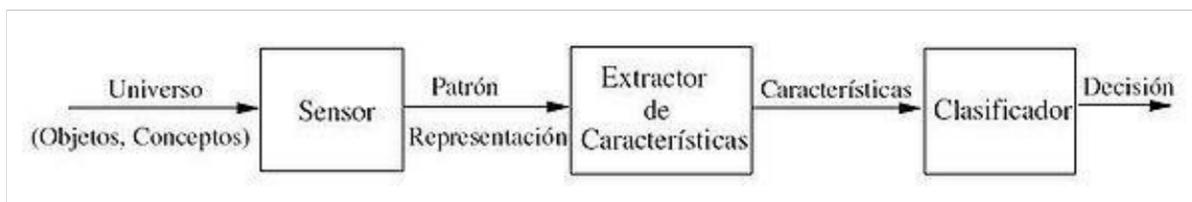
5.1.9.1 Partes del problema

- Aislar los objetos (segmentación).
- Extraer las características que discriminan los objetos.
- Construir el sistema de clasificación definiendo una distancia que permita medir que tan próximo es el elemento a cada una de las clases y en este sentido clasificar x en la clase cuya distancia se acerque más.

Un esquema general de un sistema de reconocimiento de patrones podría ser el siguiente:

Figura 55

Esquema de un sistema de reconocimiento de patrones



Nota. Datos tomados de (Wikiwand, 2022).

Por tal motivo, cuando se intenta resolver algún problema que implique el reconocimiento de patrones se deben tener en cuenta las siguientes etapas:

- 1. Adquisición de datos:** Durante esta etapa se obtienen los datos que describen al patrón que se desea clasificar. Esto implica que, el método que se utilizará para realizar la adquisición de datos, dependerá de la naturaleza del patrón. (Gutiérrez González, 2000)

2. **Preprocesamiento:** Una vez que ya se cuenta con los datos que describen a cada uno de los patrones, se lleva a cabo una fase de preprocesamiento de información con el propósito de reducir la dimensionalidad de los datos, de tal forma que todos los patrones tengan el mismo tamaño (escala), consiguiendo un alto índice de invariabilidad al escalamiento y traslación y por consiguiente un conocimiento previo. (Gutiérrez González, 2000)
3. **Extracción de características:** Este proceso trata de reducir la cantidad de información (reducción de dimensionalidad) que representa a cada uno de los patrones, obteniendo de esta forma, un vector de características que represente de la mejor manera posible al patrón original. (Gutiérrez González, 2000)
4. **Clasificación:** Consiste en obtener una respuesta acorde al conjunto de patrones recibidos como entradas. (Gutiérrez González, 2000)

5.1.10 Reconocimiento facial

5.1.10.1.1 Visión Artificial (Visión por Computadora). La visión por computador o visión artificial es una disciplina científica de inteligencia artificial conformada por un conjunto de métodos de extracción y tratamiento de imágenes del mundo real. Las imágenes son capturadas por medio de un transductor óptico que transduce las señales análogas (como la luz) en señales digitales, dichas señales pueden ser interpretadas como información numérica y pueden ser procesadas con la ayuda de un computador.

Para efectuar los procesos de la visión artificial es necesario contar con dos elementos fundamentales, el hardware y el software, siendo el hardware el encargado de la captura y recepción de las imágenes digitales y el software el encargado de realizar todo el tratamiento y procesamiento de la información. (García Santillán & Chacua Criollo, 2019)

Esta disciplina se encarga de estudiar y construir algoritmos sofisticados que analizan e interpretan la información de alta y baja calidad, siendo capaces de efectuar tareas visuales que para una persona podrían considerarse repetitivas, complejas y abrumadoras. Gracias al desarrollo tecnológico, hoy en día la visión artificial tiene un extenso campo de aplicación, ya que puede ser implementada en dispositivos móviles y computadores que cuentan con un alto nivel de procesamiento, permitiendo la interacción entre las personas y la tecnología artificial.

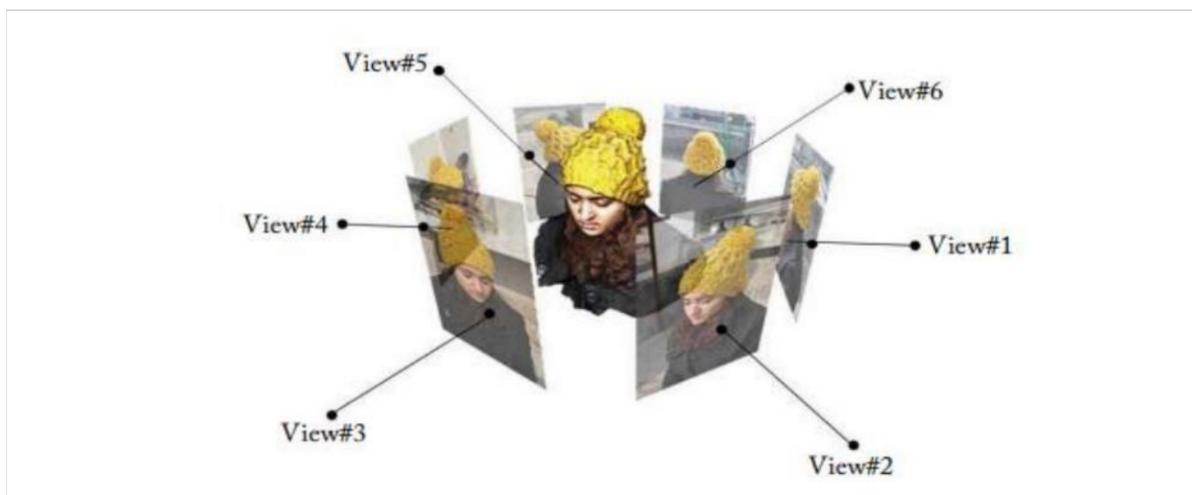
Para un humano es fácil entender el mundo que lo rodea a través de la visión, pudiendo comprender la tridimensionalidad de los objetos y sus estructuras, pero para una máquina es un proceso sumamente complicado, ya que debe recuperar, interpretar y entender objetos tridimensionales de una imagen de referencia bidimensional. Un ejemplo muy particular es la construcción de un objeto tridimensional con referencia en la captura de un conjunto de múltiples imágenes bidimensionales de distintas vistas, pudiendo establecer un modelo tridimensional preciso del objeto.

A pesar de la complejidad de los algoritmos y los avances tecnológicos, la visión artificial sigue siendo un reto de estudio para los científicos, debido a que el nivel de

comprensión e identificación de objetos de la máquina es incomparable con el nivel de interpretación visual de los humanos. (García Santillán & Chacua Criollo, 2019)

Figura 56

Construcción de un modelo 3D con base en un conjunto de imágenes bidimensionales



Nota. Datos tomados de (García Santillán & Chacua Criollo, 2019)

Tabla 25

Comparativa entre Visión Humana y Visión Artificial

Sistema Humano	Sistema Artificial
Mejor reconocimiento de objetos.	Mejor midiendo magnitudes físicas.
Mejor adaptación a situaciones imprevistas.	Mejor para la realización de tareas rutinarias.
Utilización de conocimiento previo.	Mejor en tareas de procesamiento de imágenes digitales de bajo nivel.
Mejor en tareas de procesamiento de imágenes visuales de alto nivel.	

Nota. Datos tomados de (García Santillán & Chacua Criollo, 2019).

5.1.10.1.2 Aplicaciones de la Visión Artificial. Existe una amplia gama de aplicaciones donde se puede implementar la visión artificial, en la Tabla 26

Aplicaciones de la Visión Artificial. se denotan las principales aplicaciones de acuerdo con el área de implementación.

Tabla 26

Aplicaciones de la Visión Artificial.

Área de Producción	Aplicación
Control de Calidad	Inspección de productos (papel, aluminio, acero, etc.). Identificación de piezas. Etiquetados (fechas de caducidad, etc). Inspección de circuitos impresos. Control de calidad de los alimentos (naranjas, etc.).
Robótica	Control de soldaduras. Guiado de robots (vehículos no tripulados). Robots con capacidades avanzadas.
Biomedicina	Análisis de imágenes de microcopia (virus, células, proteínas). Resonancias magnéticas, tomografías, genoma humano.
Astronomía	Exploración del espacio.
Reconocimiento de caracteres	Control de cheques, inspección de textos.
Control de tráfico	Matrículas de coches. Tráfico viario.
Meteorología	Predicción del tiempo.
Agricultura	Interpretación de fotografías aéreas. Control de plantaciones.
Militares	Seguimiento de objetivos. Vigilancia por satélites.
Vigilancia y Seguridad	Conteo de personas. Rastreo de personas y vehículos. Sistema de control de acceso. Seguimiento de trayectorias.

Nota. Datos tomados de (García Santillán & Chacua Criollo, 2019).

5.1.10.1.3 Captura de Imagen de la Cámara e Imagen Digital – Composición. Para entender cómo está constituida una imagen digital es necesario entender cómo es el proceso de captura y recolección de la imagen, como se mencionó en la definición de Visión Artificial, las imágenes digitales principalmente son capturadas por medio de un transductor óptico, (entiéndase como transductor, el sensor encargado de recolectar los haces de luz que son reflejados de los objetos), existen diferentes tipos de transductores ópticos que de acuerdo con el tamaño de su área, permiten o no un aumento en el nivel de incidencia de la luz, logrando establecer una relación de proporcionalidad directa entre estos dos factores, mejorando la calidad y el tamaño de la imagen digital.

Cuando el transductor captura los haces de luces, convierte las señales análogas en señales digitales que son interpretadas, formateadas y almacenadas por un microcontrolador; los formatos de imagen digital más utilizados son los: RAW, JPG, PNG y GIF. Existen muchos tipos de formatos, pero cada formato de imagen digital cumple con unas condiciones y características que la hacen apto para ser utilizado en tareas específicas.

De esta forma se puede definir que la imagen digital es una abstracción y representación del mundo real en una imagen bidimensional constituida a partir de una matriz numérica de datos binarios. Existen dos tipos de imágenes digitales, en primer lugar tenemos las imágenes de mapa de bits que están formadas por unidades de medida llamadas píxeles, un píxel es la unidad homogénea de una imagen digital, se codifica mediante un conjunto de bits con una longitud determinada, por lo general un píxel tiene un tamaño de 1 byte, es decir, de 8 bits, admitiendo por cada píxel valores de hasta 256 variaciones de color, con un rango de 0 a 255.

El tamaño de representación del color del pixel puede variar, por lo que también puede ser representado por 3 bytes (24 bits), es decir, una variación de color de 2^{24} colores, igualmente puede ser representado por 4 bytes (32 bits), es decir, una variación de color de 2^{32} colores, pero con la diferencia que tiene la misma cantidad de color que la representación de 3 bytes pero utiliza el byte restante para efectos de transparencia. En segundo lugar, tenemos las imágenes vectoriales, un gráfico vectorial o imagen vectorial está constituido por objetos geométricos, definidos por atributos matemáticos que establecen la posición, la forma, el color, grosor de línea, tamaño, entre otros. Una imagen vectorial es muy distinta a una imagen basada en mapa de bits, debido a que el principio fundamental de una imagen vectorial es la escalabilidad de la imagen sin pérdida de calidad, a su vez, es posible ajustar los parámetros de la imagen vectorial según se desee, ya sea estirar, comprimir, escalar, reducir, etc.

Las imágenes digitales basadas en mapa de bits están constituidas por 3 canales de color, el rojo, el verde y el azul, que en inglés estaría determinado por sus siglas como: RGB (Red, Green & Blue). El color de los canales parte de la naturalidad del ojo humano de percibir estos tres colores, ya que las células receptoras (conos y bastones), están naturalmente diseñados para interpretar esta gama de ondas de color, es por eso que, al combinar estos tres canales con un nivel de intensidad definido por cada canal, se determina un color específico. (Udemy, 2022)

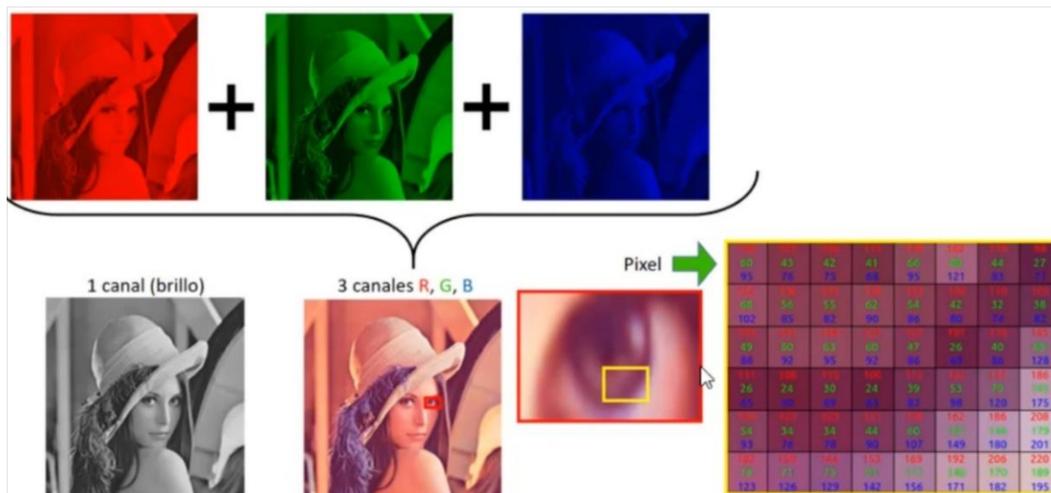
Como se aprecia en la Figura 57

Composición y representación de los canales RGB y Escala de grises de una imagen basada en mapa de bits. la imagen basada en mapa de bits, está determinada por los canales RGB, a su vez,

para tonalidades blanco y negro, se tiene un canal por separado. De igual forma, una sección de la imagen está constituida por múltiples píxeles, donde cada pixel está determinado por la combinación de los tres colores con valores distintos de 0 a 255, gracias a esto se genera el tono específico en cada sección de la imagen. (Udemy, 2022)

Figura 57

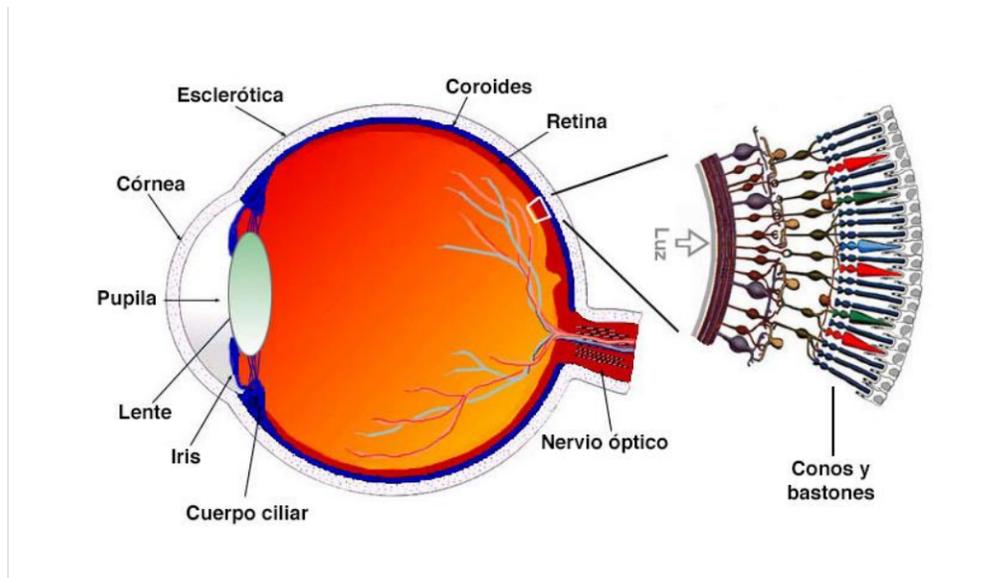
Composición y representación de los canales RGB y Escala de grises de una imagen basada en mapa de bits



Nota. Datos tomados de (Udemy, 2022).

Figura 58

Estructura interna del ojo humano



Nota. Datos tomados de (Udemy, 2022).

Teniendo en cuenta que una imagen de mapa de bits está constituida por un conjunto de números dentro de una matriz, es posible analizar el comportamiento y determinar lo que se aprecia en la imagen de acuerdo con cálculos matemáticos de dicho conjunto de números. Entre más píxeles contenga una imagen de mapa de bits, es posible llevar a cabo un mejor análisis de estudio con el algoritmo de reconocimiento artificial, debido a que se puede extraer una mayor cantidad de información. (Udemy, 2022)

5.1.10.1.4 Detección de Rostros. El proceso de detección de rostros especifica el área de una imagen digital donde se logra apreciar el rostro de una persona, lo que significa que las áreas que no contienen parte del rostro son descartadas. Logrando así, obtener una nueva imagen con la silueta del rostro identificado. Para llevar a cabo el proceso de detección de rostros, existen una gran variedad de algoritmos. (Castaño Saavedra & Alonso Sierra, 2019)

Basados en el principio de que solo se hace necesario detectar el rostro de la persona, sin llevar a cabo el proceso de identificación específicamente; se crean un conjunto de patrones para las zonas que contienen parte del rostro y las que no.

5.1.11 Técnicas de Reconocimiento Facial

5.1.11.1 PCA (Principal Component Analysis). PCA es un algoritmo de reducción dimensional que permite encontrar un conjunto de vectores que representan de la mejor forma la distribución de un grupo de imágenes. Este método está basado en la transformada de Karhunen-Loeve, que se basa en un proceso estocástico no periódico a través de una serie de vectores obtenidos completamente del mismo proceso, lo que significa que el análisis de componentes principales permite representar la imagen de un rostro por medio de una base obtenida a partir de diferentes observaciones de rostros. **(Gimeo Hernandez & Morros Rubio, 2010)**

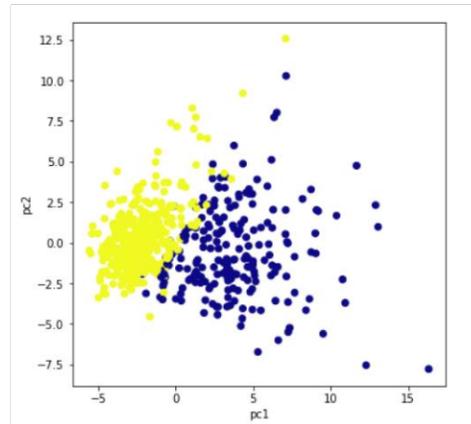
El propósito principal de PCA es representar una imagen en un sistema de coordenadas adecuadas y eficiente que reduzca el número de componentes que puede contener una imagen, tal como se puede observar en la Figura 59

Componentes principales de un conjunto de puntos bidimensionales. El primer componente $pc1$ ofrece una reducción lineal óptima de dimensión de 2D a 1D en cuanto al error cuadrático se refiere.

Figura 59

Componentes principales de un conjunto de puntos bidimensionales. El primer componente $pc1$

ofrece una reducción lineal óptima de dimensión de 2D a 1D en cuanto al error cuadrático se refiere



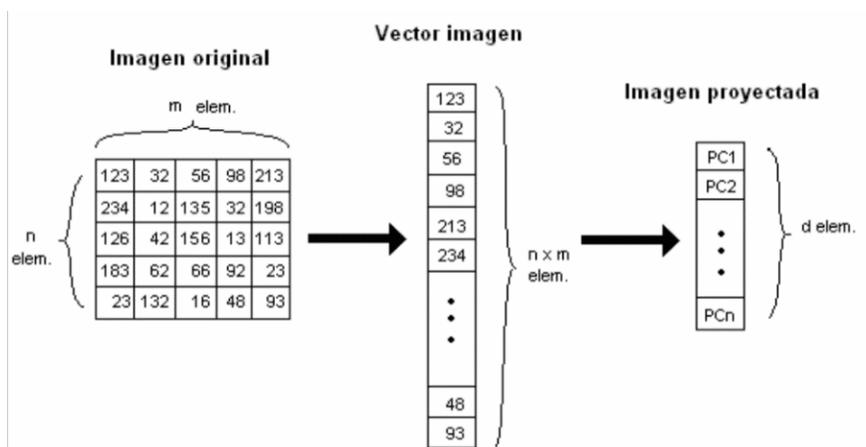
Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

La base de transformación de PCA depende de las observaciones realizadas y se compone de una serie de valores d autovalores más representativos de la matriz de covarianza, que representan las componentes principales en imágenes de diferentes rostros. La reducción de dimensiones llevada a cabo por este algoritmo es proporcional al número de autovectores implementados. Por lo tanto, la nueva imagen proyectada por PCA posee una dimensión de valor d , como se puede observar en la Figura 60

Reducción de dimensionalidad. (Gimeo Hernandez & Morros Rubio, 2010)

Figura 60

Reducción de dimensionalidad



Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

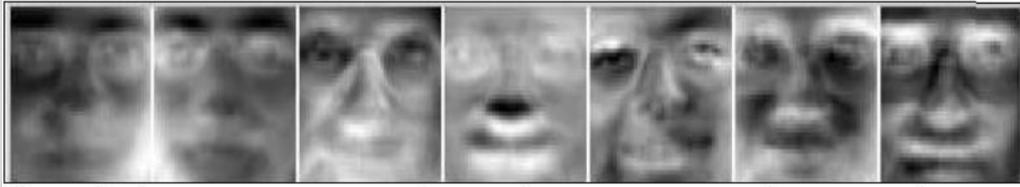
La nueva imagen está compuesta por los vectores con las componentes principales más representativas de los rostros, en otras palabras, son los ejes del nuevo espacio de dimensiones donde son proyectados los autovectores los cuales corresponden con las direcciones de máxima varianza de los datos. (Gimeo Hernandez & Morros Rubio, 2010)

La matriz de transformación o de covarianza está constituida por los autovectores relacionados a los d autovalores más significativos. Para llevar a cabo un ejemplo real se representaron los primeros siete autovectores que derivan del uso de la base de datos ORL, como se puede observar en la Figura 61

Autovectores correspondientes a los siete mayores autovalores mostrados como imágenes $p \times p = n$ (Imágenes obtenidas de la BD de rostros ORL). (Gimeo Hernandez & Morros Rubio, 2010)

Figura 61

Autovectores correspondientes a los siete mayores autovalores mostrados como imágenes $p \times p = n$ (Imágenes obtenidas de la BD de rostros ORL)



Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

La imagen promedio correspondiente a los primeros siete autovectores, la podemos observar a continuación Figura 62.

Figura 62

Imagen promedio de los primeros siete autovectores



Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

Finalmente, podemos observar las imágenes correspondientes a los siete autovalores más bajos, obtenidos de la base de datos ORL.

Figura 63

Autovectores correspondientes a los siete menores autovalores, mostrados como imágenes $pxp=n$ (Obtenidos de la BD de rostros ORL)



Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

5.1.11.2 LDA (Linear Discriminant Analysis). LDA o Linear Discriminant Analysis es una técnica de aprendizaje supervisado para clasificar datos. Es una técnica supervisada debido a que para buscar esa proyección se debe entrenar el sistema con patrones etiquetados, es importante aclarar que a diferencia de PCA, LDA no busca en ningún momento minimizar el error de representación cometido. (Universidad Politecnica de Valencia, 2017)

La idea central de LDA es obtener una proyección de los datos en un espacio de menor (o incluso igual) dimensión que los datos entrantes, con el fin de que la separabilidad de las clases sea la mayor posible. Este método tiene como objetivo convertir un problema de alta dimensión en uno de baja dimensión. Para ello LDA proyecta las imágenes en un espacio vectorial de baja dimensión de manera que el ratio entre la distancia entre clases y la distancia dentro de la clase se maximiza. De este modo se garantiza una máxima discriminación entre las clases. (Universidad Politecnica de Valencia, 2017)

Este algoritmo utiliza información de los datos, para ello maximiza la ecuación X .

$$J(w) = \frac{w^t S_B w}{w^t S_W w}$$

Donde S_b representa la matriz de dispersión de clases, mientras que S_w hace referencia a la matriz de dispersión intra clase y N_c es el número de casos dentro de la clase c y se definen de la siguiente manera.

$$S_b = \sum_c N_c (\mu_c - \bar{x})(\mu_c - \bar{x})^T$$

$$S_w = \sum_c \sum_{i \in c} (x_i - \mu_c)(x_i - \mu_c)^T$$

$$\mu_c = \frac{1}{N_c} \sum_{i \in c} x_i$$

$$\bar{x} = \frac{1}{N} \sum_i x_i = \frac{1}{N} \sum_c N_c \mu_c$$

Una vez solucionado el problema de reducción de dimensionalidad, se obtiene una matriz de proyección, las columnas de la nueva matriz resultante son las bases del nuevo espacio y son conocidas como Fisherfaces y permiten garantizar una máxima discriminación entre clases por encima de ser las que mejor describen los datos, tal como se observa en la Figura 64.

(Universidad Politecnica de Valencia, 2017)

Figura 64

Las primeras seis FisherFaces calculadas en la base de datos ORL



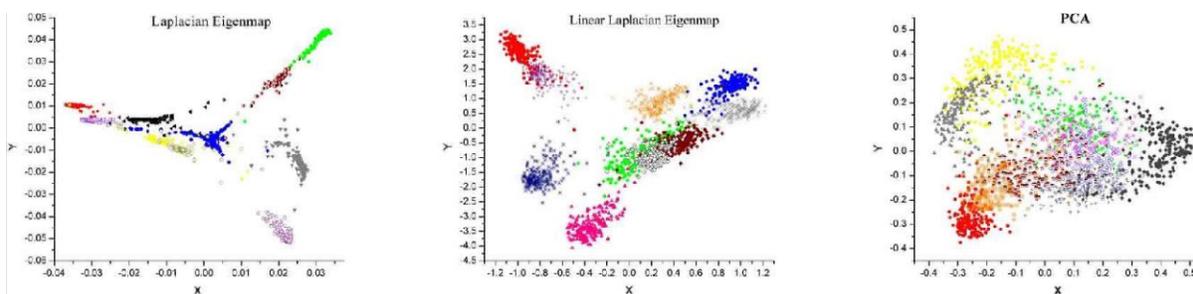
Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

LDA permite realizar una mejor descripción de los datos, sin embargo, posee una limitación al utilizar matrices de dispersión no-singulares. Esta restricción en aplicaciones de reconocimiento facial desemboca en un problema de matrices singulares, ya que en muchas ocasiones la alta dimensionalidad de las imágenes es mucho mayor que el número de imágenes. Para evitar esta inconsistencia, se utilizan diferentes aproximaciones como pseudo-inverse LDA que implementa la pseudo-inversa de la matriz de covarianza de PCA en lugar de su inversa, por otra parte, también realiza una reducción previa de los datos utilizando PCA, junto con LDA. (Gimeo Hernandez & Morros Rubio, 2010)

5.1.11.3 LPP (Locality Preserving Projections). LPP, al igual que PCA es un algoritmo lineal que se basa en la reducción de dimensionalidad de datos. Sin embargo, la principal característica de este método es una estructura local de datos, que permite identificar los datos ubicados en el espacio de dimensiones original de alta dimensionalidad en el nuevo espacio de baja dimensionalidad en el mismo lugar. Al conservar la estructura local de la información, las imágenes correspondientes a un mismo individuo siempre se encontrarán cercanas entre sí y distantes de las imágenes de otros individuos, lo que resulta en una discriminación marcada entre clases. (Gimeo Hernandez & Morros Rubio, 2010)

Figura 65

A diferencia del posicionamiento de datos entre PCA y LPP, este último conserva su estructura local de datos



Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

Para lograr conservar su estructura de datos local, LPP implementa un grafo de adyacencias el cual incorpora información de la estructura de los mismos. El grafo de adyacencia consiste básicamente en la creación de una matriz de tamaño $N \times N$, donde N representa el número de imágenes con pesos asignados de acuerdo a la posición de los elementos, es decir si los elementos i y j son vecinos o no. (Gimeo Hernandez & Morros Rubio, 2010)

La creación del grafo de adyacencia generalmente depende del campo de aplicación, ya que existe una forma determinada de creación para problemas de aprendizaje no supervisado y supervisado. En el caso de no supervisado, su creación se lleva a cabo de forma automática, aplicando métodos de búsqueda basados en vecinos como lo son *K-nearest neighbors* o *E-neighborhoods*. No obstante, para el caso supervisado la creación se lleva a cabo de forma manual a través del proceso de identificación y asignación de quienes son o no los vecinos de cada elemento. (Gimeo Hernandez & Morros Rubio, 2010)

LPP al ser un método lineal, tiene la ventaja de ser un algoritmo rápido y conveniente para aplicaciones prácticas. Además, cuenta con mapas diseñados para minimizar algún criterio objetivo procedente de las técnicas lineales clásicas y a diferencia de las técnicas de reducción

lineales clásicas, LPP es capaz de representar los datos en el nuevo subespacio de baja dimensión sin realizar el cálculo de la matriz de proyección. No obstante, este algoritmo es un poco inestable para el caso de recuperación de datos originales a partir del nuevo espacio de baja dimensionalidad, ya que el proceso se dificulta debido a la implementación de bases no ortogonales. (Gimeo Hernandez & Morros Rubio, 2010)

5.1.11.4 LBPH (Local Binary Pattern Histogram). LBPH es un método de patrones binarios locales diseñado para la descripción de texturas. Generalmente, la implementación de técnicas de reconocimiento facial basadas en descripciones locales de algunas regiones de las imágenes del rostro aporta más información que otras, por lo que los descriptores locales tienden a promediar la información que describen, aspecto que no resulta conveniente al describir rostros ya que en estos casos mantener los datos de las relaciones espaciales es importante. **(Esparza Franco, Tarazona Ospina, Sanabria Cuevas, & Velazco Capacho, 2015)**

Para lograr una descripción global, la imagen del rostro es dividida en diferentes áreas, a las cuales se les aplica un histograma con el que se obtiene el operador LBPH para realizar una descripción independiente por cada área. Estas descripciones locales son concatenadas con el propósito de construir una descripción global del rostro. (Esparza Franco, Tarazona Ospina, Sanabria Cuevas, & Velazco Capacho, 2015)

El algoritmo LBPH asigna etiquetas a cada uno de los píxeles de la imagen teniendo en cuenta la distribución de los vecinos, posteriormente, usando una máscara de tamaño determinado (8×8), recorre la imagen de manera repetitiva, seleccionando cada vez un píxel

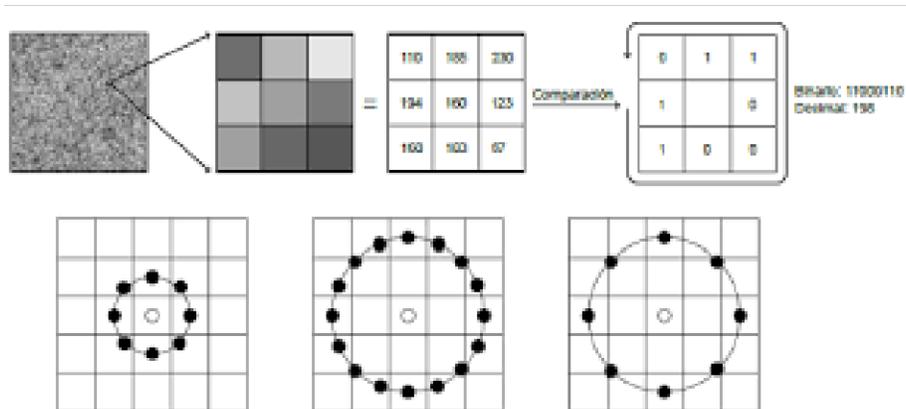
central y sus vecinos. Cada píxel central se compara con cada uno de los vecinos de forma ordenada. Se asigna un 1 cada vez que el píxel central sea menor que pixel comparado y un 0 en el caso contrario, como se muestra en la ecuación y en la Figura 66. (Esparza Franco, Tarazona Ospina, Sanabria Cuevas, & Velazco Capacho, 2015)

$$LBP = \sum_{p=0}^7 s(g_p - g_c)2^p$$

$$s(x) = \begin{cases} 1, & \text{si } x \geq 0 \\ 0, & \text{otro valor} \end{cases}$$

Figura 66

Proceso de obtención de parámetros LBPH



Nota. Datos tomados de (Esparza Franco, Tarazona Ospina, Sanabria Cuevas, & Velazco Capacho, 2015).

El número binario resultante se transforma en un número decimal que es contado en el histograma para formar la descripción. El histograma de las etiquetas de todos los píxeles se

utiliza posteriormente como una descripción de la textura de la imagen, tal como se puede observar en la ecuación . (Esparza Franco, Tarazona Ospina, Sanabria Cuevas, & Velazco Capacho, 2015)

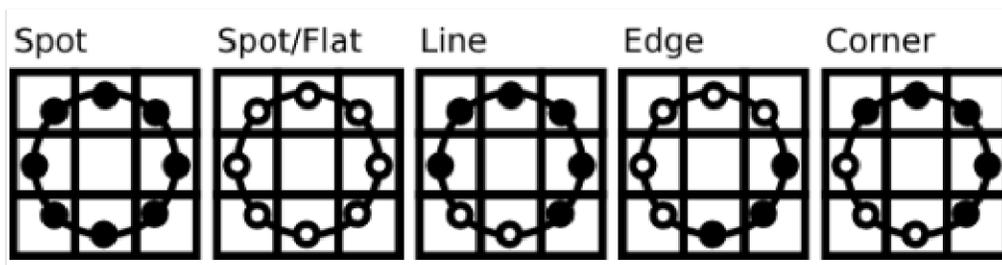
$$H_i = \sum_{xy} I = LBP(x, y) = i, \quad i = 0, \dots, n - 1$$

$$I(x) = \begin{cases} 1, & \text{si } x \text{ es verdadero} \\ 0, & \text{otro valor} \end{cases}$$

Sin embargo, luego de obtener el resultado, las investigaciones demostraron que un vecino fijo no puede codificar detalles que varían en escala. Por lo que este método, optó por usar una cantidad de vecinos variables, con el objetivo de alinear de forma aleatoria una cantidad x de vecinos en un círculo de radio variable para lograr capturar los *neighborhoods*.

Figura 67

Identificación de neighborhoods



Nota. Datos tomados de (OpenCV, 2022).

Para un punto específico (x_c, y_c) la posición del vecino (x_p, y_p) , $p \in P$ puede ser calculado por:

$$x_p = x_c + R \cos\left(\frac{2\pi p}{P}\right)$$

$$y_p = y_c + R \sin\left(\frac{2\pi p}{P}\right)$$

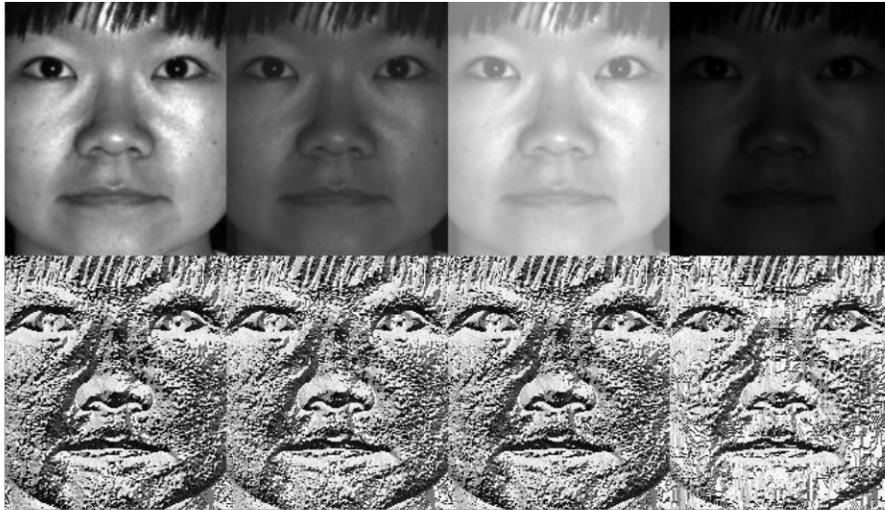
Donde R representa el radio del círculo y P el número de puntos de muestra. El operador es una extensión de los códigos LBP originales, es por eso que también se le conoce como LBP extendido o LBP circular. Si la coordenada de un punto en el círculo no corresponde con las de la imagen, el punto se interpola. Las librerías actualmente poseen diversos algoritmos que implementan algún tipo de interpolación, OpenCV usa interpolación bilineal, tal como se puede observar en la siguiente ecuación. (OpenCV, 2022)

$$f(x, y) \approx [1 - x \quad x] \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1 - y \\ y \end{bmatrix}$$

Por definición, el operador LBP es robusto frente a transformaciones monotónicas en escala de grises. Podemos verificar esto fácilmente mirando la imagen LBP de una imagen modificada artificialmente:

Figura 68

Comparación de una imagen con aplicación de LBP vs imagen artificial



Nota. Datos tomados de (OpenCV, 2022).

El algoritmo incorpora la información espacial obtenida en el modelo de reconocimiento facial. La representación propuesta por Ahonen es dividir la imagen LBP en m regiones locales y extraer un histograma de cada una. Finalmente, el vector de características optimizado espacialmente se obtiene concatenando los histogramas locales (*no fusionados*) y estos histogramas son conocidos como *histogramas de patrones binarios locales (LBPH)*. (OpenCV, 2022)

5.1.11.5 DCT (Discrete Cosine Transform). El algoritmo de DCT consiste en una transformación que representa una secuencia limitada de pasos como la suma de una serie de funciones cosenoidales variando entre diferentes frecuencias. Es una técnica común en aplicaciones de procesamiento de señales digitales, especialmente en el proceso de reconocimiento facial, desde la compresión de audio e imágenes hasta variaciones espectrales para la solución numérica de ecuaciones diferenciales. (Gimeo Hernandez & Morros Rubio, 2010)

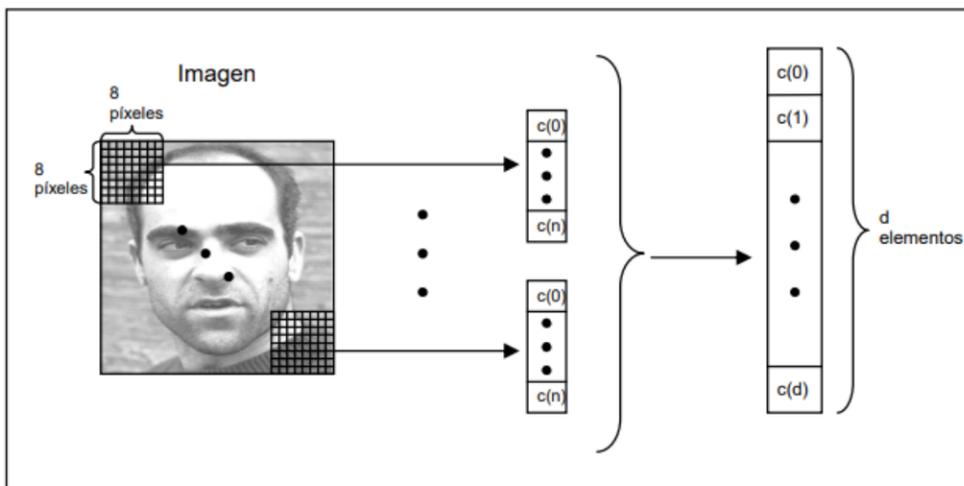
DCT a diferencia de PCA, no requiere de un entrenamiento de imágenes del mismo tipo que utilizará, solo realiza la transformación directamente de las imágenes, es decir, la base de la transformación es independiente de las imágenes, además de esto es un algoritmo de bajo coste computacional comparado con PCA. (Gimeo Hernandez & Morros Rubio, 2010)

5.1.11.6 DCT – Bloques. DCT por bloques lleva a cabo una mezcla de los algoritmos basados en apariencia y las técnicas basadas en modelos, específicamente hace uso de la misma metodología de su predecesor DCT, sin embargo, dicha metodología es aplicada de forma distinta. (Gimeo Hernandez & Morros Rubio, 2010)

Este algoritmo, aplica DCT sobre bloques de tamaño 8×8 píxeles de la imagen. Una vez se ha calculado la DCT sobre cada uno de los bloques, se selecciona un determinado número de coeficiente por bloque con el fin de crear la nueva representación de la imagen, tal como lo podemos observar en la Figura 69. (Gimeo Hernandez & Morros Rubio, 2010)

Figura 69

Esquema de funcionamiento de DCT por bloques

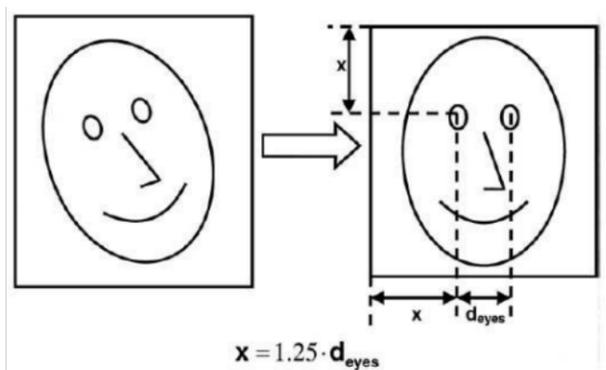


Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

DCT por bloques implementa de forma parcial los sistemas basados en modelos ya que requiere de la ubicación de ciertas características del rostro. En este caso para el funcionamiento ideal de un sistema DCT por bloques, se requiere que las imágenes sean recortadas y alineadas. Por lo tanto, para lograr la aplicación de forma correcta, previamente deben identificarse la posición de los ojos y alinear las imágenes en base a un determinado criterio, tal como se puede observar a continuación. (Gimeo Hernandez & Morros Rubio, 2010)

Figura 70

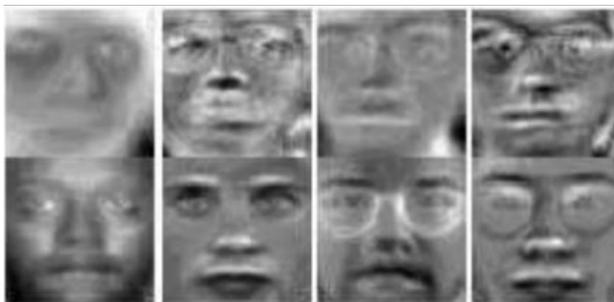
Alineación de las caras usando DCT por bloques



Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

Figura 71

Autocaras calculadas sin alinear (superior) y alineadas (inferior)



Nota. Datos tomados de (Gimeo Hernandez & Morros Rubio, 2010).

5.1.12 Técnicas de Reconocimiento Facial Específicas (OpenCV)

OpenCV es una librería de visión por computador de código abierto, desarrollada por Intel desde el año de 1999. La librería está enfocada en el procesamiento de imágenes en tiempo real, además permite su aplicabilidad en todo tipo de proyectos gracias a la inclusión de implementaciones libres de patentes de los últimos algoritmos desarrollados de visión por computadora. (OpenCV, 2022)

Además, OpenCV está publicado bajo licencia BSD (Berkeley Software Distribution), pudiendo ser utilizada en proyectos académicos y en productos comerciales, implementando C, C++, Python y Android.

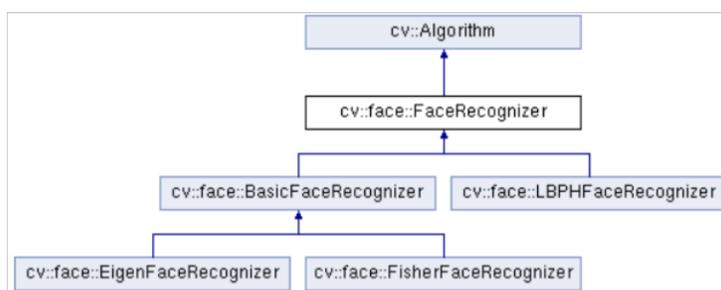
A partir de la versión 2.4, OpenCV integró una nueva clase llamada FaceRecognizer que permite realizar el reconocimiento facial, permitiendo que la experimentación y la implementación del reconocimiento facial pueda ser efectuado de forma inmediata. (OpenCV, 2022)

Actualmente OpenCV tiene a su disposición tres algoritmos de reconocimiento:

- Eigenfaces (EigenFaceRecognizer::create)
- Fisherfaces (FisherFaceRecognizer::create)
- Histogramas de patrones binarios locales (LBPHFaceRecognizer::create)

Figura 72

Estructura de clases de los algoritmos de reconocimiento facial de la librería OpenCV



Nota. Datos tomados de (OpenCV, 2022).

Debido a la disponibilidad de los algoritmos de reconocimiento facial incorporados en la librería de OpenCV, se ha tomado como referencia de estudio e implementación la utilización de estos algoritmos durante el desarrollo del proyecto.

5.1.12.1 Reconocimiento Facial con OpenCV. El reconocimiento facial es una tarea sencilla para los humanos, se ha logrado evidenciar en varios experimentos que bebés de uno a tres días de nacimiento pueden distinguir rostros entre personas conocidas. Comprender el proceso de reconocimiento facial de los humanos es actualmente un campo de investigación, por lo que se desconocen parcialmente cómo se realiza el procedimiento. Existen muchas especulaciones de cómo podría ser el proceso de reconocimiento facial, se cree que nace del análisis de las características internas (boca, ojos, nariz) o externas del rostro (forma de la cabeza, línea del cabello).

El estudio del proceso de reconocimiento facial, ha llevado a los investigadores a preguntarse cómo el cerebro analiza y codifica una imagen. David Hubel y Torsten Wiesel pudieron demostrar que el cerebro posee dentro de su estructura, células nerviosas especializadas que responden a estímulos determinados frente a las características de una escena, como las líneas, bordes, ángulos o movimiento.

Debido a que los humanos aprecian el mundo a su alrededor de una forma concreta, la corteza visual (la zona encargada del cerebro de decodificar la percepción y convertirla en visión) debe de alguna manera combinar las fuentes de información percibidas y convertirlas en

información con patrones útiles. El reconocimiento facial automático se trata de extraer esas características significativas de una imagen, ponerlas en una representación útil y realizar algún tipo de clasificación en ellas. (OpenCV, 2022) (Salud Ocular, 2022)

El enfoque más intuitivo de reconocimiento facial está basado en las características geométricas de la cara, en el surgimiento de los primeros sistemas automatizados de reconocimiento de rostros, se determinaron puntos de marcaje de acuerdo con la ubicación de las características principales del rostro (ojos, nariz, boca, oídos, etc.), estos puntos se utilizaron para construir un vector de características que determinara la distancia entre ellos, su ángulo, posición, etc. El reconocimiento se llevó a cabo al calcular la distancia euclídea entre los vectores de características de una imagen de prueba y una imagen de referencia. Debido a la naturaleza de este método, es posible realizar reconocimientos en ambientes con cambios de iluminación, pero realizar el registro preciso de los puntos de marcaje es una tarea complicada incluso para los algoritmos más modernos. (OpenCV, 2022)

Las investigaciones sobre reconocimiento geométrico pudieron llegar a la conclusión que, a pesar de utilizar vectores de características con múltiples dimensiones en grandes conjuntos de datos, las características basadas en geometría no logran satisfacer las necesidades de la información para realizar el reconocimiento facial. (OpenCV, 2022)

Posteriormente surgieron varios métodos que tuvieron un enfoque distinto para realizar el reconocimiento facial, uno de estos métodos fue Eigenfaces que planteaba realizar el reconocimiento bajo una metodología holística, es decir, analizando las propiedades en conjunto

y no solo las partes que la componen. Eigenfaces determina que una imagen facial es un área del espacio de una imagen de alta dimensión, donde se encuentra una representación de menor dimensión, haciendo que la clasificación se vuelva más sencilla. (OpenCV, 2022)

Para encontrar el subespacio de menor dimensión se realiza el análisis de componentes principales (**PCA**), identificando los ejes con la máxima varianza. Realizar este tipo de transformaciones es óptimo para la reconstrucción, pero no tiene en cuenta ningún tipo de etiqueta de clase, por lo que los ejes con varianza máxima no contienen información discriminatoria que permita realizar una clasificación. Debido a este inconveniente, se hace necesario aplicar una proyección específica de clase a través del análisis discriminante lineal (**LDA**) para el reconocimiento facial. La idea principal es minimizar la varianza dentro de una clase, mientras se maximiza (al mismo tiempo) la varianza entre clases. (OpenCV, 2022)

Actualmente existen multitud de métodos para extraer características locales, donde se describen las regiones locales de una imagen, evitando la alta dimensionalidad de los datos de entrada. Las características extraídas son teóricamente más robustas frente a cambios en la iluminación, la oclusión parcial y el tamaño de muestreo pequeño. Los algoritmos más utilizados para extraer las características locales son: **LBP** (Local Binary Patterns), **DCT** (Discrete Cosine Transform) y **Gabor Wavelets**. Determinar cuál es la mejor forma de preservar la información espacial cuando se aplica un método de extracción de características locales es una de las preguntas de investigación vigente, debido a que toda información espacial es información que podría ser condicionalmente útil. (OpenCV, 2022)

5.1.12.2 Bases de datos de rostros. Para realizar el reconocimiento facial es necesario contar con una base de datos de rostros con diferentes características de acuerdo a la necesidad, en la actualidad existen un sin fin de bases de datos especializadas, por lo que no es necesario construir una base de datos de rostros propia. OpenCV puntualiza que existen tres bases de datos interesantes para su aplicabilidad.

- ***AT&T Facedatabase:*** Esta base de datos también es conocida como ORL Facedatabase, la base de datos está constituida por 10 imágenes de 40 personas distintas. Las imágenes de algunas personas fueron tomadas en diferentes momentos, con condiciones ambientales diferentes donde se veía una variación en la iluminación, las expresiones faciales y los accesorios o detalles del rostro. A su vez, se tomaron controladamente contra un fondo homogéneo de color oscuro y con las personas en posición frontal erguida con una tolerancia de movimiento lateral.
- ***Yale Facedatabase A:*** Esta base de datos también es conocida como Yalefaces, el modelo A de Yalefaces es un conjunto de datos más apropiado para experimentos iniciales, debido que el problema de reconocimiento es más difícil. Esta base de datos está constituida por 11 imágenes en escala de grises de 15 personas (14 hombres, 1 mujer) con un tamaño de 320px x 243px. Presenta cambios en las condiciones de luz, modificando la luz central, luz izquierda y luz derecha, además, expresiones faciales y utilización de accesorios como gafas.
- ***Extended Yale Facedatabase B:*** Esta base de datos está constituida (en su versión recordada) por 2414 imágenes de 38 personas diferentes. La idea principal de esta base de datos es establecer características que son difíciles de determinar dentro de un sistema

de reconocimiento, por lo que aspectos como la iluminación son netamente robustos y la variación de expresividades, oclusión y demás características, son casi invariables.

Yalefaces B es un conjunto de datos muy amplio, por lo que para experimentos pequeños es más recomendable utilizar Yalefaces A o la base de datos de AT&T. La versión extendida de Yalefaces B nace precisamente de la integración de estas dos bases de datos.

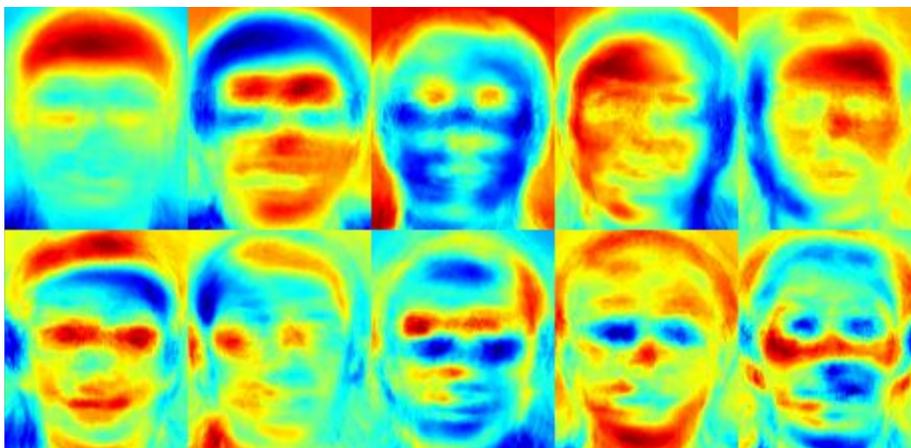
5.1.12.3 Aplicabilidad de las técnicas de Reconocimiento Facial en OpenCV. La

documentación de OpenCV brinda de forma práctica la aplicabilidad de cada uno de los métodos de reconocimiento facial implementados dentro de ella.

5.1.12.3.1 PCA (Eigenfaces) en OpenCV. En el ejemplo documentado, se utiliza un mapa de colores jet para apreciar la distribución de valores en la escala de grises distribuidas en Eigenfaces. En él se puede apreciar que las Eigenfaces no solo determinan los rasgos faciales, sino también el cambio en la iluminación de las imágenes. Se puede apreciar que la Eigenface número cuatro tiene un cambio de luz, distinto a la Eigenface número 5. (OpenCV, 2022)

Figura 73

Eigenfaces con mapa de colores jet OpenCV



Nota. Datos tomados de (OpenCV, 2022).

Como se mencionó anteriormente es posible reconstruir una cara a partir de una aproximación dimensional baja. Para una reconstrucción es necesario tener aproximadamente mínimo unos 50 Eigenvectores, ya que, si se tuvieran menos Eigenvectores que esta cifra, no se podrían extraer características faciales importantes. Para garantizar una buena reconstrucción, tomando como referencia la base de datos de AT&T, es necesario tener aproximadamente 300 vectores propios. La variación del número de Eigenfaces necesarios para realizar el reconocimiento y que sea exitoso, depende en gran medida de los datos de entrada, por lo que difiere según la base de datos que esté siendo utilizada. (OpenCV, 2022)

Figura 74

Resultado de n Eigenvectores en OpenCV



Nota. Datos tomados de (OpenCV, 2022).

5.1.12.3.2 LDA (Fisherfaces) en OpenCV. Debido a que PCA solo se encarga de maximizar la variación total de los datos sin tener en cuenta valores discriminantes al desechar componentes, hace que se pierda información de utilidad. Suponiendo que la variación de los datos es generada por una fuente de luz externa, al utilizar PCA no se tiene en cuenta esta discriminación en los componentes identificados por lo que las muestras proyectadas se unifican y el proceso de clasificación se vuelve nulo. (OpenCV, 2022)

Para solucionar el inconveniente de PCA, el estadístico Sir RA Fisher, inventó el método LDA, el cual permite una reducción de dimensionalidad específica por clase, para demostrar su

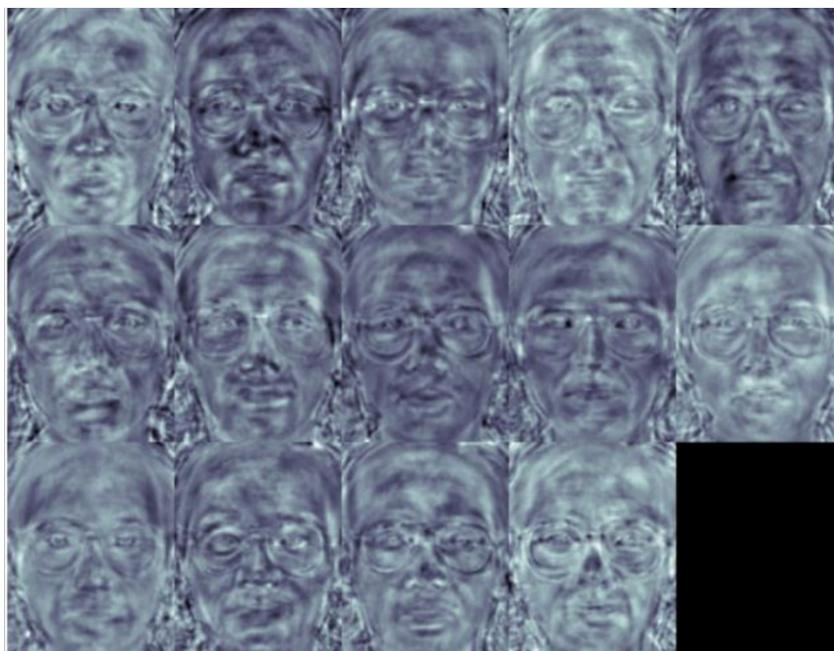
aplicabilidad Fisher la implementó exitosamente para clasificar flores en su artículo de 1936: “*El uso de medidas múltiples en problemas taxonómicos*”. (OpenCV, 2022)

LDA maximiza la proporción de dispersión entre clases y dentro de las clases, para encontrar la combinación de características de separación, en vez de maximizar la dispersión general. Se podría resumir que LDA nace de una idea sencilla, agrupar estrechamente las mismas clases, mientras las diferentes clases están lo más alejadas posibles entre sí de la representación de dimensiones inferiores. (OpenCV, 2022)

Utilizando LDA con Yale Facedatabase A y teniendo en cuenta que cada Fisherface tiene la misma longitud que una imagen original, se puede representar como una imagen.

Figura 75

Muestra de Fisherfaces



Nota. Datos tomados de (OpenCV, 2022).

Fisherfaces aprende una matriz de transformación específica de clase, haciendo que no capturen la iluminación de forma tan precisa como Eigenfaces, pero a diferencia, LDA determina las características faciales de discriminación de las personas. A su vez, el método de Fisherfaces se ve afectado en su optimización de acuerdo con los datos de entrada, es decir, si se utiliza en un sistema donde solo se tienen almacenadas imágenes bien iluminadas y se intenta realizar un reconocimiento de rostros en escenas con mala iluminación, probablemente el resultado sea que se encuentren los componentes incorrectos, debido a que estas características no predominan en una imagen mal iluminada, gracias a que no tuvo la oportunidad de aprender ese tipo de iluminación. (OpenCV, 2022)

Tanto PCA como LDA permiten una reconstrucción de la imagen proyectada, pero debido a que solo extrae las características de distinción entre las personas, no es posible reconstruir una imagen parecida a la original. (OpenCV, 2022)

Al proyectar la imagen de muestra en cada de las Fisherfaces, se obtendrá una buena visibilidad que describe cada una de las características de esta. Para el ojo humano la diferencia entre cada una puede ser sutil, a pesar de esto, se logra apreciar un mínimo de diferencias. (OpenCV, 2022)

Figura 76

Fisherfaces con OpenCV



Nota. Datos tomados de (OpenCV, 2022).

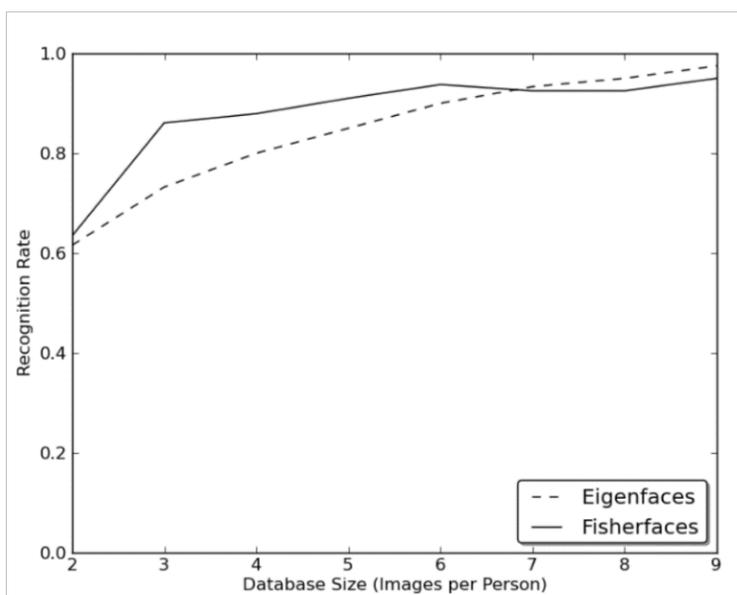
5.1.12.3.3 LBPH en OpenCV. Eigenfaces como Fisherfaces están fundamentados bajo un enfoque holístico para el reconocimiento facial, Eigenfaces trata los datos como un vector en lugar de un espacio de imagen de alta dimensión, reduciendo a un subespacio de menor dimensión los componentes, por ende, pierde información que posiblemente pudiera ser útil, además maximiza la dispersión total, generando posibles problemas de varianza debido a fuentes externas que hace que los componentes con varianza máxima en algunas de las clases sean inútiles para la clasificación. (OpenCV, 2022)

Debido a esto se utiliza LDA para preservar datos discriminatorios, teniendo en cuenta que se hace bajo un entorno controlado, donde la diferencia de iluminación y demás factores externos se mantiene estable. En la vida real entenderíamos que no es perfecta, por lo que, es

imposible garantizar el control de los factores externos en los ajustes de las imágenes capturadas. Dado el caso de que solo se tenga una imagen por cada persona, las estimaciones de covarianza para el subespacio pueden ser totalmente equivocadas, por ende, el reconocimiento también. (OpenCV, 2022)

Figura 77

Tasa de reconocimiento de rango 1 del método Eigenfaces y Fisherfaces en la base de datos AT&T Facedata



Nota. Datos tomados de (OpenCV, 2022).

Entonces, para la obtención de tasas adecuadas de reconocimiento se necesitan al menos 8 imágenes por persona, por lo que el método de Fisherfaces se volvería impráctico en menos de esta cantidad. Algunas investigaciones se concentraron en extraer las características locales de las imágenes, para evitar dichos inconvenientes. La idea principal es evitar mirar la imagen como un vector de alta dimensión, sino más bien describir las características locales de un objeto

determinado, haciendo que se extraiga implícitamente una baja dimensionalidad. A pesar de esta premisa, las imágenes no solo sufren alteraciones de iluminación sino también de escalado y rotación, por lo que su descripción local debe ser aún más robusta frente a estas alteraciones. (OpenCV, 2022)

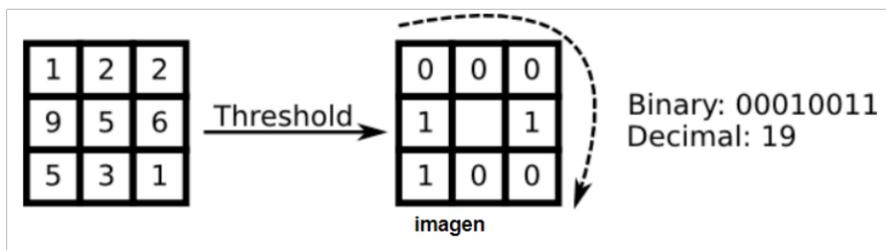
La idea principal de los patrones binarios locales es resumir la estructura local de una imagen comprando cada píxel con el vecindario de píxeles a su alrededor. Si el píxel central en relación con la referencia del umbral de píxeles es mayor o igual a su vecino, se indica como un 1 o un cero si se sobrepasa o no, por lo que se obtendrá un sistema binario. (OpenCV, 2022)

Es decir que con 8 píxeles circundantes al píxel central se tendrá una posibilidad 2^8 combinaciones, llamadas códigos LBP.

Por ejemplo:

Figura 78

Operador LBP descrito con una vecindad fija de 3x3



Nota. Datos tomados de (OpenCV, 2022).

5.1.13 Comparativa de las principales técnicas de Reconocimiento Facial

Tabla 27

Comparativa de las principales técnicas de reconocimiento facial

Técnicas	Ventajas	Desventajas
Clasificador de Bayes	<p>El modelo Native Bayes se deriva de la teoría matemática clásica, tiene una base matemática sólida y una eficiencia de clasificación estable.</p> <p>Posee una mayor velocidad al entrenar y consultar grandes números.</p> <p>Funciona bien en datos a pequeña escala, puede manejar tareas de clasificación múltiple y es adecuado para entrenamiento incremental (entrena nuevas muestras en tiempo real)</p> <p>La interpretación de los resultados es fácil de entender.</p>	<p>Requiere calcular la probabilidad previa.</p> <p>Presenta inconvenientes en la tasa de error de las decisiones de clasificación.</p> <p>Es sensible a la forma de expresión de los datos de entrada</p>
Redes neuronales	<p>La RNA es paralela, por lo que pueden ser usadas en tiempo real.</p>	<p>Tiempo de aprendizaje elevado. Esto depende de dos factores: primero si se incrementa la cantidad de patrones a identificar o clasificar y segundo si se requiere mayor flexibilidad o capacidad de adaptación de la red neuronal para reconocer patrones que sean sumamente parecidos, se deberá invertir más tiempo en lograr que la red converja a valores de pesos que representan lo que se quiera enseñar.</p>

Técnicas	Ventajas	Desventajas
	Debido a que una RNA almacena la información de forma redundante, ésta puede seguir respondiendo de manera aceptable aun si se daña parcialmente.	Complejidad de aprendizaje para grandes tareas, cuantas más cosas se necesiten que aprenda una red, más complicado será enseñarle.
	Una RNA crea su propia representación de la información en su interior, descargando al usuario de esto.	Elevada cantidad de datos para el entrenamiento, cuanto más flexible se requiere que sea la red neuronal, más información tendrá que enseñarle para que realice de forma adecuada la identificación.
Redes neuronales	Una RNA puede manejar cambios no importantes en la información de entrada, como señales con ruido u otros cambios en la entrada.	No permite interpretar lo que se ha aprendido, la red por sí sola proporciona una salida, un número, que no puede ser interpretado por ella misma, sino que se requiere de la intervención del programador y de la aplicación en sí para encontrarle un significado a la salida proporcionada.
	La red puede seguir llevando a cabo su función aun cuando parte de la red haya sido destruida.	
	Son simples de entender y de interpretar.	Los aprendices de árbol de decisión tienden al sobre entrenamiento, especialmente cuando el número de características predictivas es alto.
Árboles de decisión	No requiere una preparación de los datos demasiado exigente (aunque la implementación de Scikit-Learn no soporta valores nulos).	Son inestables: cualquier pequeño cambio en los datos de entrada puede suponer un árbol de decisión completamente diferente.
	Se puede trabajar tanto con variables cuantitativas como cualitativas.	No se puede garantizar que el árbol generado sea el óptimo.
	Utiliza un modelo de caja blanca: la respuesta del algoritmo es fácilmente justificable a partir de la lógica booleana implementada en él.	Hay conceptos que no son fácilmente aprendibles pues los árboles de decisión no son capaces de expresarlos con facilidad (como el operador XOR).

Técnicas	Ventajas	Desventajas
Máquinas de soporte vectorial	Tienen una gran capacidad de generalización y esto los lleva a brindar gran robustez en el reconocimiento.	La técnica SVM implementan reglas de decisión complejas
	Eficaz en espacios de grandes dimensiones.	Los SVM no proporcionan directamente estimaciones de probabilidad, se calculan usando una validación cruzada quintuple.
	Utiliza un subconjunto de puntos de entrenamiento en la función de decisión, por lo que también es eficiente en memoria.	
PCA	Reduce la dimensión del problema.	No tiene presente la discriminación entre las clases del problema.
	Proporciona un mayor rendimiento, eficacia y rapidez en la extracción de características.	
LDA	Reduce la dimensión del problema.	La clasificación de nuevas observaciones no es muy eficiente a medida que se incrementa el número de variables del modelo.
	Contempla la discriminación entre las clases del problema.	Requiere que se especifiquen los grupos del conjunto de entrenamiento del modelo con clases prefijadas.
	El vector discriminante ofrece una indicación de que variables tienen más poder Discriminante.	Se ve afectado frente a factores externos.
LBPH	Permite reconocer texturas.	Si no se extiende el operador en un vecindario fijo, no puede codificar detalles que difieren en escala.
	Puede capturar detalles muy finos en las imágenes.	

Nota. Datos tomados de

5.2 Construcción del prototipo funcional de reconocimiento facial

El prototipo de reconocimiento facial se desarrolló con la intención de abrir un espacio investigativo en el área de las ciencias de visión asistida por computadora, que permitiera no solo el desarrollo del software, sino también la implementación de artefactos tecnológicos como apoyo a la reducción de la brecha de inseguridad dentro de la infraestructura de la universidad, afianzando la prevención y la restricción de accesos a fuentes de información sensibles. Para el desarrollo del sistema se implementó una metodología de prototipos definida por una serie de iteraciones cuyo propósito era la construcción de una versión funcional aproximada del software final, fundamentada en el análisis y nivel de prioridad de los requerimientos de software.

5.2.1 Iteración 1

5.2.1.1 Requerimientos

Inicialmente se llevó a cabo un estudio del estado del arte que permitiera ampliar el panorama conceptual conforme a las necesidades de la problemática, identificando los requerimientos funcionales y no funcionales del sistema, para establecer una base de herramientas que aseguraran el desarrollo de una versión funcional del software.

Como resultado del estudio previo, se estructuraron los requerimientos de acuerdo con su categoría y nivel de prioridad, definiéndolos como requerimientos iniciales, requerimientos funcionales y requerimientos no funcionales.

Muchos de estos requerimientos no se pudieron cumplir dentro de las primeras iteraciones, debido a que eran requerimientos de implementación o no se contaban con los recursos para satisfacerlos.

5.2.1.1.1 Requerimientos iniciales. Considerando que la metodología de prototipos está orientada al desarrollo ágil, se definieron una serie de requerimientos iniciales para identificar desde una perspectiva global los aspectos más importantes para el desarrollo y ejecución del proyecto.

Tabla 28

Requerimientos Iniciales del Sistema

No.	Requerimientos Iniciales del Sistema	Nivel de Prioridad		
		Alta	Media	Baja
Requerimientos de Interfaz				
RIS1	El sistema debe permitir detectar rostros y reconocerlos en tiempo real.	x		
RIS2	El sistema debe permitir el registro de las personas según sus rasgos biométricos faciales.	x		
RIS3	El sistema debe permitir capturar las imágenes de vídeo a través de una cámara de videovigilancia.		x	
RIS4	El sistema debe permitir la renderización de la imagen de vídeo de forma efectiva a través de la unidad de procesamiento gráfico.	x		

No.	Requerimientos Iniciales del Sistema	Nivel de Prioridad		
		Alta	Media	Baja
RIS5	El sistema debe permitir conectividad las 24 horas del día a la red eléctrica.	x		
Requerimientos de Funcionamiento				
RIS6	El sistema debe permitir realizar el reconocimiento facial para la identificación y registro de las personas conocidas.	x		
RIS7	El sistema debe permitir realizar el reconocimiento facial para la identificación y registro de las personas desconocidas.	x		
RIS8	El sistema debe permitir realizar la detección y reconocimiento facial de las personas dentro del entorno no supervisado del Edificio de Ingeniería.	x		
RIS9	El sistema debe permitir obtener el vídeo en tiempo real de la cámara de seguridad del circuito cerrado de televisión.	x		
Requerimientos de Estados				
RIS10	El sistema debe permitir actividad permanente de procesamiento de las imágenes de vídeo obtenidas por la cámara.	x		
Requerimientos Físicos				
RIS11	El sistema debe ser discreto y estar ubicado en un lugar dónde no interfiera con las actividades de las personas que hacen uso del recinto.		x	
RIS12	La cámara de seguridad debe estar ubicada en un lugar estratégico que permita el enfoque y la captura adecuada de las personas que ingresan al recinto, favoreciendo a la reducción de agentes externos que alteren la calidad de la imagen.		x	
RIS13	La cámara de seguridad debe estar ubicada a una altura idónea para permitir la captura del rostro de las personas de manera adecuada y así realizar el proceso de identificación.		x	

Nota. Realizado por los autores.

5.2.1.1.2 Requerimientos funcionales. Los requerimientos funcionales se definieron de acuerdo con las tareas administrativas y de uso particular a las cuales podían acceder los usuarios en el sistema.

Tabla 29

Requerimientos funcionales

No.	Requerimientos Funcionales	Nivel de Prioridad		
		Alta	Media	Baja
Requerimientos Operacionales				
RU1	El sistema debe implementarse en las instalaciones de la Universidad Francisco de Paula Santander, específicamente en el Edificio de Ingeniería.		x	
RU2	El sistema debe permitir la adquisición de información para el entrenamiento y construcción del clasificador de aprendizaje automático.	x		
RU3	El sistema debe permitir almacenar los rostros de las personas en directorios de forma ordenada dentro del servidor para realizar el entrenamiento.	x		
Requerimientos de Usuario				
RU4	El sistema debe permitir al administrador realizar las operaciones CRUD para gestionar los accesos al sistema (Nombre, Apellido, Código, Documento, Correo, Estado, Acción editar, Acción habilitar, Acción inhabilitar).		x	
RU5	El sistema debe permitir al administrador realizar las operaciones CRUD para gestionar los módulos del sistema (Nombre, Ruta, Estado, Acción editar, Acción habilitar, Acción inhabilitar).		x	

No.	Requerimientos Funcionales	Nivel de Prioridad		
		Alta	Media	Baja
RU6	El sistema debe permitir al administrador realizar las operaciones CRUD para gestionar los perfiles del sistema (Nombre, Estado, Acción editar, Acción habilitar, Acción inhabilitar).		x	
RU7	El sistema debe permitir al administrador Realizar las operaciones CRUD para gestionar las personas que interactuarán con el sistema (Nombre, Apellido, Código, Documento, Correo, Perfil, Acción editar, Acción capturar rostro, Acción identificar persona).		x	
RU8	El sistema debe permitir al administrador Realizar las operaciones CRUD para gestionar la asociación del módulo con el perfil del sistema (Nombre, Perfil, Estado, Acción habilitar, Acción inhabilitar, Buscar módulo, Buscar perfil).		x	
RU9	El sistema debe permitir al administrador visualizar el streaming de las cámaras que identifican a las personas, sea monitoreo, registro de usuarios o verificación de rasgos biométricos de los usuarios en el sistema.		x	
RU10	El sistema debe permitir el acceso a los usuarios por medio de credenciales (Código, Contraseña).		x	
RU11	El sistema debe permitir a los usuarios indirectos identificar imágenes del rostro sin ser supervisados.		x	

Nota. Realizado por los autores.

5.2.1.1.3 Requerimientos no funcionales. Los requerimientos no funcionales se estructuraron en cuatro categorías: diseño, hardware, software y eléctrico. Los requerimientos de diseño hacen referencia a la categoría de diseño de la arquitectura de red e interconectividad, los de hardware al tipo de hardware necesario para el correcto funcionamiento del sistema, los de software a las herramientas necesarias para realizar el reconocimiento facial y los eléctricos a la disponibilidad eléctrica para que el sistema funcione en jornada continua.

Tabla 30

Requerimientos no funcionales

No.	Requerimientos de No Funcionales	Nivel de Prioridad		
		Alta	Media	Baja
Requerimientos de Diseño				
RA1	La cámara debe estar instalada sobre una pared dentro de la zona estratégica del Edificio de Ingeniería, apuntando correctamente para la obtención de imágenes.		x	
RA2	La conexión Ethernet debe ser extendida a través del sistema de cableado estructurado del Edificio de Ingeniería mediante una red de área local.		x	
RA3	El servidor debe estar ubicado en un lugar adecuado para su correcta ventilación.		x	
Requerimientos de Hardware				
RA4	El sistema requiere una unidad central de procesamiento que permita el tratamiento y procesamiento de imágenes en tiempo real.	x		
RA5	El sistema requiere una unidad de procesamiento gráfico que permita el tratamiento y renderizado de imágenes en tiempo real.	x		
RA6	El sistema requiere una fuente de poder gama alta de más de 500 vatios, para suplir el suministro energético necesario para	x		

No.	Requerimientos de No Funcionales	Nivel de Prioridad		
		Alta	Media	Baja
	el correcto funcionamiento de todos los componentes electrónicos del servidor.			
RA7	Se requiere una fuente de poder con certificación, que garantice un margen de error mínimo en el suministro de la energía.	x		
RA8	Se requiere una tarjeta madre con conexión Ethernet para la transmisión y comunicación de datos a través de la red.	x		
RA9	Se requiere una unidad de procesamiento gráfico que permita el entrenamiento de modelos de aprendizaje automático y profundo.	x		
RA10	Se requiere una unidad de procesamiento gráfico para realizar el procesamiento de cómputo en paralelo.	x		
RA11	Se requiere una cámara de seguridad de alta resolución para garantizar el proceso de reconocimiento facial.		x	
RA12	Las imágenes deben ser procesadas rápidamente por la unidad central de procesamiento.	x		
RA13	Se requiere una unidad de almacenamiento capaz de almacenar volúmenes considerables de información para el almacenamiento de las fotos capturadas.	x		
RA14	La unidad central de procesamiento del servidor debe ser compatible con cualquier arquitectura de los sistemas operativos disponibles en la actualidad.		x	
RA15	La unidad central de procesamiento debe ser de última generación.	x		
RA16	La unidad central de procesamiento debe como mínimo ofrecer una velocidad de frecuencia sobre 2.5Ghz.	x		
RA17	Si no es posible tener una unidad de procesamiento gráfico independiente, la unidad central de procesamiento debe contar con la integración dentro del mismo chip.	x		
RA18	Se requiere de una cámara de seguridad con conexión Ethernet que garantice la transmisión de imágenes bajo el protocolo de internet (IP).	x		

No.	Requerimientos de No Funcionales	Nivel de Prioridad		
		Alta	Media	Baja
Requerimientos de Software				
RA19	Se requiere de un sistema operativo, lenguaje de programación y herramientas de desarrollo de código abierto.		x	
RA20	Se requiere un lenguaje de programación que sea compatible con la librería de OpenCV.	x		
RA21	Se requiere que el sistema permita la ejecución de visión artificial en tiempo real dentro del servidor.	x		
RA22	Se requiere de un sistema operativo que sea capaz de ejecutar multiprocesos rápidamente por medio de los diferentes hilos que lo integran.	x		
RA23	Se requiere un software que permita el uso dinámico de los recursos de la unidad de procesamiento gráfico en el entrenamiento de los modelos de aprendizaje.	x		
RA24	Se requiere un software de base de datos que permita acceder a los datos de forma efectiva y permita la implementación CRUD.	x		
RA25	Se requiere compatibilidad de software con bibliotecas de aprendizaje automático.	x		
RA26	Se requiere compatibilidad de software con tecnología de cómputo en paralelo.		x	
Requerimiento Eléctrico				
RA27	El sistema debe estar conectado permanentemente a la red eléctrica.		x	

Nota. Realizado por los autores.

5.2.1.2 Diseño y construcción

5.2.1.2.1 Hardware. Con base en los requerimientos de hardware planteados y considerando que el procesamiento gráfico de imágenes de video y el entrenamiento del sistema requerían de la implementación de un hardware robusto, se llevó a cabo un análisis sobre los componentes esenciales, como la CPU (Unidad Central de Procesamiento), la GPU (Unidad de Procesamiento Gráfico), la PSU (Unidad de Fuente de Alimentación) y la cámara de videovigilancia, con el objetivo de garantizar el funcionamiento integral e idóneo del sistema.

CPU. En el mercado de hardware existe gran variedad de CPUs, por tal motivo se hizo necesario establecer criterios de selección para evaluar cada opción disponible. Dentro de los criterios se encontraron: la cantidad de núcleos, el número de hilos de procesamiento y la frecuencia de trabajo; criterios de que según el tipo de arquitectura del CPU y optimización del software, permitían ejecutar tareas con un mayor o menor nivel de eficiencia.

Con base en los criterios de selección establecidos previamente y el análisis de las diferentes alternativas de CPUs existentes en el mercado, considerando la relación costo-funcionalidad, se optó por la implementación del CPU Ryzen 9 3900X, que contaba con las siguientes especificaciones:

Tabla 31

Especificaciones técnicas del CPU

RYZEN 9 3900X	
Número de núcleos de CPU	12
Número de hilos	24

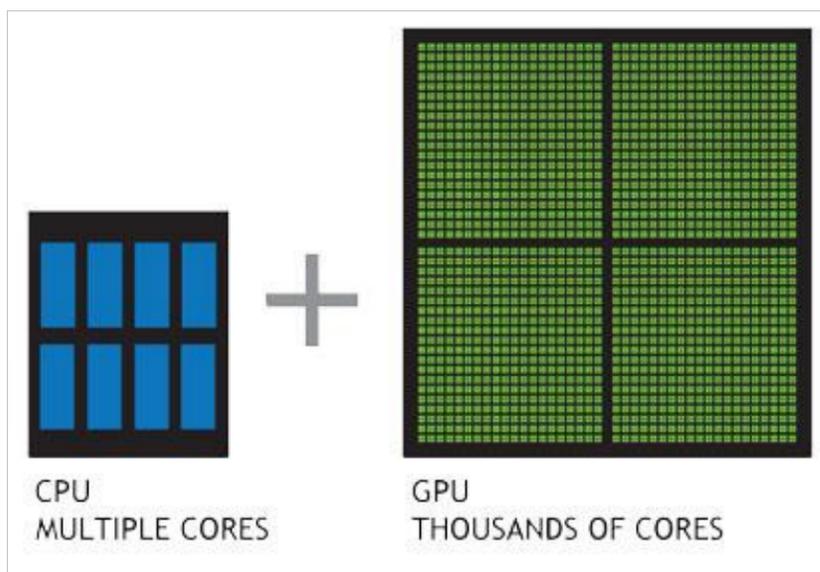
RYZEN 9 3900X	
Reloj base	3.8Ghz
Reloj de aumento máximo	Hasta 4.6GHz

Nota. (AMD, 2021)

GPU. “Una forma sencilla de comprender la diferencia entre una GPU y una CPU es comparar la forma en que procesan las tareas. Una CPU tiene unos cuantos núcleos optimizados para el procesamiento en serie secuencial, mientras que una GPU cuenta con una arquitectura en paralelo enorme que consiste de miles de núcleos más pequeños y eficaces, y que se diseñaron para resolver varias tareas al mismo tiempo”. (NVIDIA, NVIDIA, 2021)

Figura 79

Comparativa de núcleos entre el CPU y el GPU



Nota. (NVIDIA, NVIDIA, 2021)

Por este motivo, la utilización de GPUs dentro del procesamiento de visión por computadora se convierte en una herramienta casi indispensable para la optimización de los procesos de aprendizaje, reduciendo con efectividad el tiempo de entrenamiento. Dentro de la ejecución del proyecto, debido a costos, la selección de la GPU se realizó con base en la disponibilidad adquisitiva de los investigadores, permitiendo que el proyecto pudiera ser desarrollado con una tarjeta gráfica NVIDIA RTX 2060 Super con tecnología CUDA; tecnología desarrollada por NVIDIA para el procesamiento en paralelo, idónea para el entrenamiento de los clasificadores y el procesamiento de las imágenes de vídeo, debido a que ofrece a los desarrolladores la aceleración de las aplicaciones informáticas, aprovechando todos los recursos de la tarjeta gráfica. (NVIDIA, NVIDIA, 2021)

Tabla 32

Especificaciones técnicas del GPU

NVIDIA RTX 2060 Super	
Chip Gráfico	
Modelo GPU	TU106
Arquitectura	Turing
Fabricación	12nm TSMC
Núcleos Secundarios	272 núcleos tensoriales
Núcleos Terciarios	32 trazado de rayos
Uds. Textura	136
Uds. Renderizado	64
Frecuencia Base	1470 Mhz

NVIDIA RTX 2060 Super	
Memoria	
Memoria	8 GB
Tipo de Memoria	GDDR6 (14GHz)
Interfaz	256 bits

Nota. (NVIDIA, NVIDIA Corporation, 2021)

PSU. Escoger una buena fuente de alimentación era trascendental para el buen funcionamiento de los componentes de hardware mencionados, por tal motivo, se escogió una fuente de alimentación Corsair de RM850X con certificación 80 PLUS Gold, garantizando el suministro estable de energía para los componentes del hardware.

Tabla 33

Especificaciones técnicas del PSU

Corsair RM850X	
Continuous power W	850Watts
80 PLUS Efficiency	Gold
PSU Form Factor	ATX

Nota. (CORSAIR, 2021)

Cámara de videovigilancia. Para la selección de la cámara de videovigilancia se priorizaron aspectos que se adaptaran a las necesidades del proyecto, por tal motivo se seleccionó una cámara que permitiera conectarse directamente a la red local a través de conexión Ethernet bajo el protocolo RTSP (Protocolo de Transmisión en Tiempo Real) y que ofreciera una calidad de vídeo óptima para el análisis de las imágenes de visión por computadora, debido a que, entre mejor fuera la calidad de la imagen captada, se garantizaría un mejor estudio de su información.

Tabla 34

Especificaciones técnicas de la cámara de videovigilancia

DS-2CD21893G0-I	
Cámara	
Sensor de Imagen	1/2.5" Progressive Scan CMOS
Iluminación Mínima	Color: 0.01 lux @(F1.2, AGC ON), 0.028 lux @(F2.0, AGC ON), 0 lux with IR
Velocidad de Obturación	1/3 s to 1/100,000 s
Obturador Lento	Yes
Wide Dynamic Range	120dB
Día y Noche	IR Cut Filter
Ajuste de Ángulo	Pan: 0° to 355°, tilt: 0° to 75°, rotate: 0° to 355°
Lente	
Enfoque	Fixed
Lente	2.8/4/6/8 mm
FOV	

DS-2CD21893G0-I

2.8 mm, horizontal FOV: 102°, vertical FOV: 53°, diagonal FOV: 124°, 4 mm, horizontal FOV: 79°, vertical FOV: 42°, diagonal FOV: 95°, 6 mm, horizontal FOV: 50°, vertical FOV: 28°, diagonal FOV: 59°, 8 mm, horizontal FOV: 40°, vertical FOV: 25°, diagonal FOV: 53°

Apertura	F2.0
Montura del Lente	M12
Tipo de Apertura	Fixed

Vídeo

Max. Resolución	3840 × 2160
Main Stream	50Hz: 12.5 fps (3840 × 2160), 20fps (3072 × 1728), 25 fps (2560 × 1440, 1920 × 1080, 1280 × 720), 60Hz: 15 fps (3840 × 2160), 20fps (3072 × 1728), 30 fps (2560 × 1440, 1920 × 1080, 1280 × 720)
Sub Stream	50Hz: 25fps (640 × 480, 640 × 360, 320 × 240), 60Hz: 30fps (640 × 480, 640 × 360, 320 × 240)
Third Stream	50Hz: 25fps (1280 × 720, 640 × 360, 352 × 288), 60Hz: 30fps (1280 × 720, 640 × 360, 352 × 240)
Compresión de Video	Main stream: H.265/H.264, Sub-stream: H.265/H.264/MJPEG, Third stream: H.265/H.264
Vídeo Bitrate	32 Kbps to 16 Mbps
Tipo H.264	Main Profile/High Profile
Tipo H.265	Main Profile
H.264 +	Main stream supports
H.265 +	Main stream supports
Codificación de Video	H.264 and H.265 encoding support

Red

Vista en Vivo Simultánea

API	ONVIF (PROFILE S, PROFILE G), ISAPI
------------	-------------------------------------

DS-2CD21893G0-I

Protocolos	TCP/IP, ICMP, HTTP, HTTPS, FTP, DHCP, DNS, DDNS, RTP, RTSP, RTCP, PPPoE, NTP, UPnP, SMTP, SNMP, IGMP, 802.1X, QoS, IPv6, Bonjour
Usuario/Operador	Up to 32 users,3 levels: Administrator, Operator and User
Almacenamiento en Red	Support Micro SD/SDHC/SDXC card (128G), local storage and NAS (NFS,SMB/CIFS), ANR
Cliente	iVMS-4200, Hik-Connect, iVMS-5200, iVMS-4500
Navegador Web	Plug-in required live view:;IE8+, Chrome 41.0-44, Firefox 30.0-51, Safari 8.0-11,Plug-in free live view:;Chrome 45.0+, Firefox 52.0+

Imagen

Interruptor Día/Noche	Day/Night/Auto/Schedule/Triggered by Alarm In (-S)
Recorte Objetivo	No
DNR	3D DNR
Mejora de la Imagen	BLC/3D DNR /HLC
Configuración de Imagen	Saturation, brightness, contrast, sharpness, AGC, and white balance are adjustable by client software or web browser

Interfaz

Interfaz de Comunicación	1 RJ45 10M/100M self-adaptive Ethernet port
---------------------------------	---

Certificaciones

Protección	IK10, IP67
-------------------	------------

Nota. (Hangzhou Hikvision Digital Technology Co., Ltd., 2021)

De igual forma, se tuvieron en cuenta aspectos esenciales a como el tamaño del sensor y el tipo de apertura del diafragma, considerando que, a mayor tamaño del área del sensor y

apertura del diafragma, aumenta la incidencia de los rayos de luz en el sensor, permitiendo capturar una imagen de mayor resolución y calidad en distintas condiciones de luminosidad.

Figura 80

DS-2CD21893G0-I



Nota. Vista fotográfica de la cámara de videovigilancia DS-2CD21893G0-I. (Hangzhou Hikvision Digital Technology Co., Ltd., 2021)

Hardware adicional. Es importante resaltar que además del hardware principal, se requería de hardware adicional congruente con el funcionamiento de la máquina de cómputo, como la placa base, memorias RAM, discos de almacenamiento y refrigeración del sistema

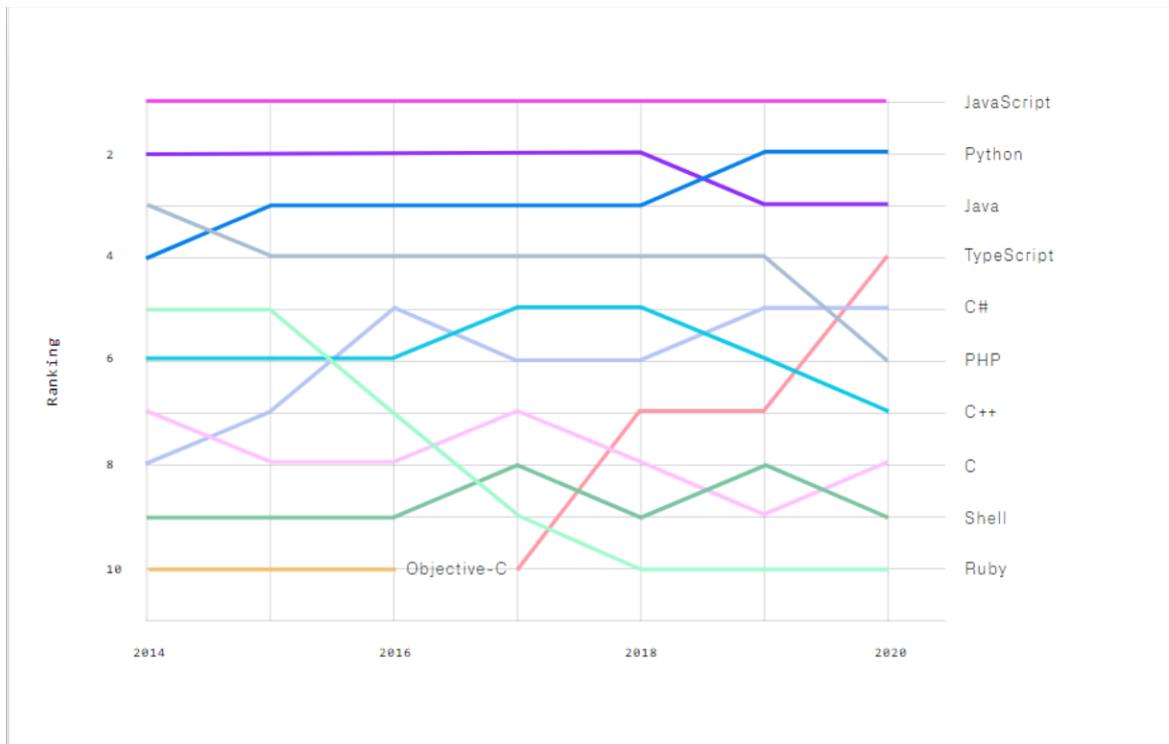
5.2.1.2.2 Software.

Lenguaje de programación. “La selección del lenguaje de programación se ha analizado en diversos estudios que van desde los años cincuenta hasta 2012” (Stefik, et al., 2015). “aunque se puede seguir un proceso formal y bien estructurado para la selección del lenguaje empleado en el desarrollo de aplicaciones que involucran inteligencia artificial, ya existen multitud de experiencias y estudios que avalan las ventajas de Python como candidato bien posicionado”. (García Monsálvez, 2017)

Python es un lenguaje de programación poderoso y fácil de aprender. Cuenta con estructuras de datos eficientes y de alto nivel, así como también con un enfoque simple pero efectivo a la programación orientada a objetos. La elegante sintaxis de Python y su tipado dinámico, junto con su naturaleza interpretada y su alta compatibilidad con bibliotecas de visión por computadora y aprendizaje profundo, hacen de este lenguaje ideal para scripting y desarrollo rápido de aplicaciones de inteligencia artificial funcionales sobre la mayoría de las plataformas, (Serrano, Universidad Don Bosco, 2019) así mismo, además de contar con una amplia comunidad en la web y documentación, se encuentra posicionado en el ranking de los lenguajes de programación más utilizados a lo largo de los años, desde el 2014 hasta el 2020, según el listado Octoverse lanzado por GitHub, (GitHub, 2020) por tal motivo, Python fue seleccionado como el lenguaje de programación sobre el que se desarrollaría todo el prototipo.

Figura 81

Top languages over the years - Lenguajes de programación a lo largo de los años



Nota. El gráfico muestra la usabilidad de los lenguajes de programación desde el 2014 hasta el 2020. (GitHub, Inc., 2021)

Librerías. Durante el desarrollo del software se implementaron diversas librerías, las cuales serán descritas a lo largo del proyecto, sin embargo, dentro de esta primera iteración las librerías que se decidieron abarcar fueron OpenCV y Numpy.

OpenCV (Open Source Computer Vision Library), es una librería de código abierto con licencia BSD que permite su uso particular o comercial. (Noguera, 2012) OpenCV incorpora funciones de programación dirigidas primordialmente a la detección por computadora en tiempo real y visión artificial (Noguera, 2012), cuenta con disponibilidad para múltiples plataformas y lenguajes de programación como Python, Android, Java, C+ y C. (Ipanaqué, 2020)

Estas características convertían a OpenCV en una opción viable para el desarrollo del sistema, debido a que era una herramienta necesaria para la lectura y manipulación de imágenes de vídeo e incorporaba una amplia gama de más de 2500 algoritmos optimizados para el procesamiento de imágenes y aprendizaje de máquina, (Noguera, 2012) y Numpy permitía realizar todo tipo de operaciones matriciales para la manipulación de los píxeles de las imágenes de vídeo.

Interfaces. Para el desarrollo de las interfaces dentro de la primera iteración se tuvieron en cuenta las librerías Tkinter y Qt, especializadas en el desarrollo de interfaces para Python con enfoque de desarrollo de software de escritorio.

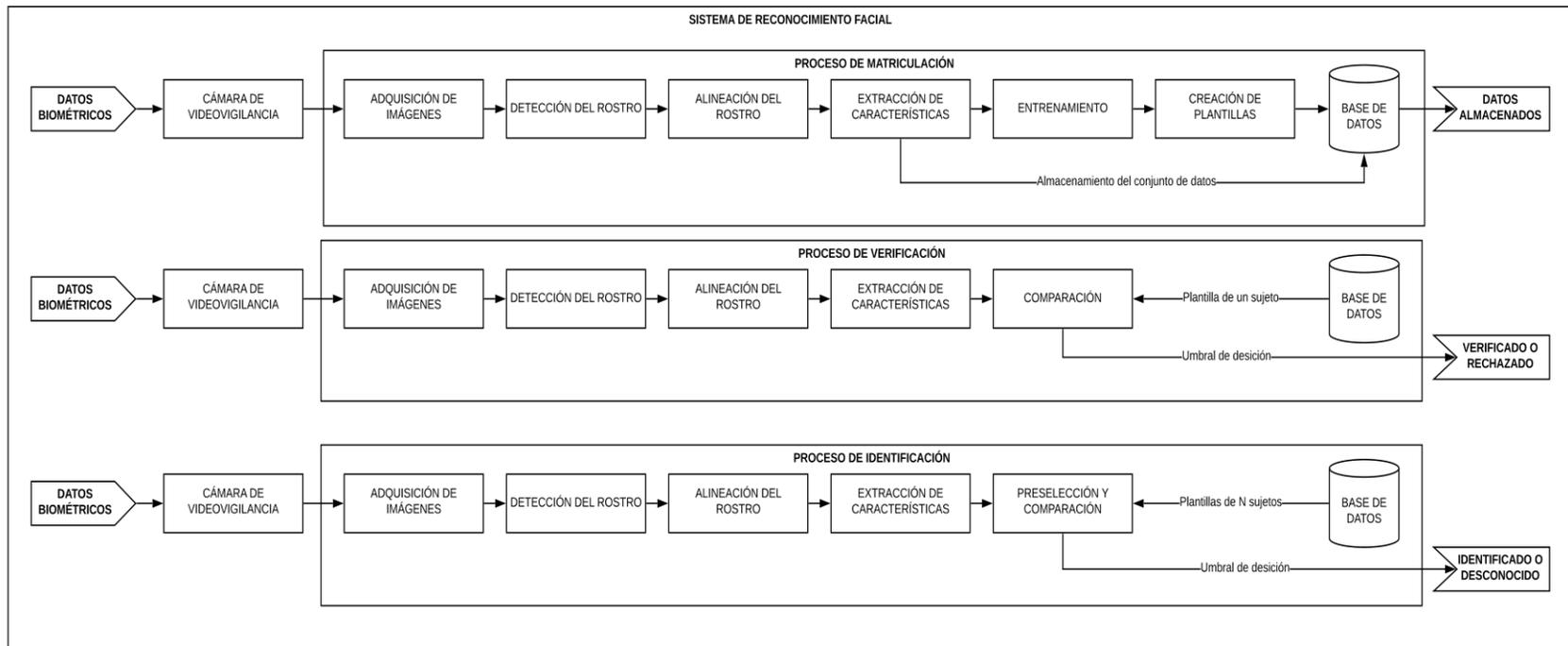
A pesar de que Tkinter era una librería que ofrecía todos los aspectos fundamentales del GUI (Interfaz Gráfica de Usuario), no permitía una modificación 100% de todos los elementos que se requerían, por tal motivo se descartó y se planteó el uso de Qt como GUI para el desarrollo del proyecto, pero en el momento de su implementación, se realizó una reestructuración de las herramientas de desarrollo, optando por herramientas que permitieran el 100% de las modificaciones de la interfaz y fueran más intuitivas para los desarrolladores, por tal motivo, el proyecto se orientó hacia la implementación de herramientas de desarrollo web.

5.2.1.2.3 Arquitectura. En la trayectoria de desarrollo del sistema, la arquitectura fue estructurándose según los requerimientos del sistema y las herramientas que se estipulaban en cada iteración, pero se concretó sobre las bases de los sistemas de reconocimiento biométricos, donde se capturaban los rasgos biométricos por medio de un transductor, se procesaba la información, se extraían las características de acuerdo con el enfoque biométrico, se asociaban los datos recolectados con la información correspondiente y se realizaba el proceso de validación en relación entre el rasgo biométrico y el sujeto.

Con base en esto, se estructuró una arquitectura de reconocimiento facial fundamentada en el estado del arte, constituida por los procesos de matriculación, verificación e identificación; procesos que originaban un resultado de ejecución necesario para la integralidad del sistema de reconocimiento facial.

Figura 82

Arquitectura del sistema de reconocimiento facial



Nota. Realizado por los autores.

El diagrama muestra los procesos estructurales del sistema de reconocimiento facial, el proceso de matriculación, el proceso de verificación y el proceso de identificación. Realizado por los autores.

Proceso de matriculación. El proceso de matriculación se definió con el propósito de captar los rasgos biométricos de una persona a través de la implementación de diferentes algoritmos de visión asistida por computadora, realizando un tratamiento de imágenes de vídeo mediante el uso de un sistema inteligente de videovigilancia.

Por consiguiente, mediante la ejecución de la tarea de detección del rostro se buscaría establecer la ubicación facial de la persona en la imagen utilizando un clasificador predefinido y se capturarían series de fotos redimensionadas y estandarizadas para la construcción del conjunto de datos, buscando así una asociación entre el conjunto de datos y las características biométricas con la información personal registrada en el sistema al ejecutarse la tarea de entrenamiento, generando como resultado un modelo integral de identificación con todos los rasgos faciales de las personas registradas.

Proceso de verificación. Una vez ejecutado el proceso de matriculación y creado el modelo de datos, sería necesario realizar un proceso de verificación para validar que los datos registrados en el sistema correspondieran con los rasgos biométricos asociados a la persona, debido a que, si no se realizara este procedimiento, existiría una vulnerabilidad en la seguridad del sistema, presentando inconsistencias en el reconocimiento facial que generarían como resultados falsos positivos, es decir, el reconocimiento de personas cuyos rasgos faciales no corresponderían con su identidad.

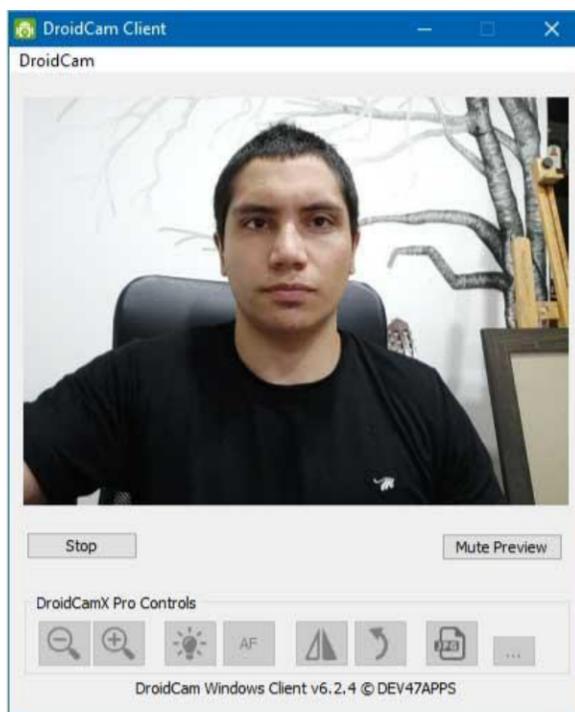
El proceso de verificación brindaría como resultado dos estados, el estado de “verificado” y el estado de “rechazado”; siendo “verificado” cuando el sistema reconociera a la persona en relación con sus rasgos biométricos y “rechazado” cuando no pudiera reconocer a la persona correctamente, estableciendo así que, si se presentaba dicha situación, se debería realizar de nuevo el proceso de matriculación, capturando un nuevo conjunto de imágenes del rostro de la persona y re entrenando el sistema. Una vez verificada la relación de los rasgos biométricos de la persona, el sistema ya estaría listo para poder realizar el proceso de identificación.

Proceso de identificación. Una vez efectuada la matriculación y verificación de la persona, se podría ejecutar el proceso de identificación, similar al de verificación, diferenciándose en el resultado, ya que este proceso daría como resultado la identificación de la persona dentro del modelo N de personas registradas en el sistema y notificaría sobre el conocimiento o desconocimiento de las personas captadas por la cámara de videovigilancia.

Desarrollo. En la primera iteración, a pesar de realizar un análisis sobre las cámaras de videovigilancia del mercado, no se contaba con una que permitiera realizar el desarrollo bajo un entorno real, por tal motivo se utilizaron herramientas que solventaran dicha necesidad, implementado así DroidCam como fuente de vídeo, debido a que DroidCam permitía convertir el teléfono celular en una cámara web para el PC, haciendo uso de un componente cliente y una aplicación móvil, emulando la cámara de videovigilancia a través de conexiones USB, WLAN o WiFi. (Dev47Apps, 2021)

Figura 83

Cliente de DroidCam – Vista de la cámara

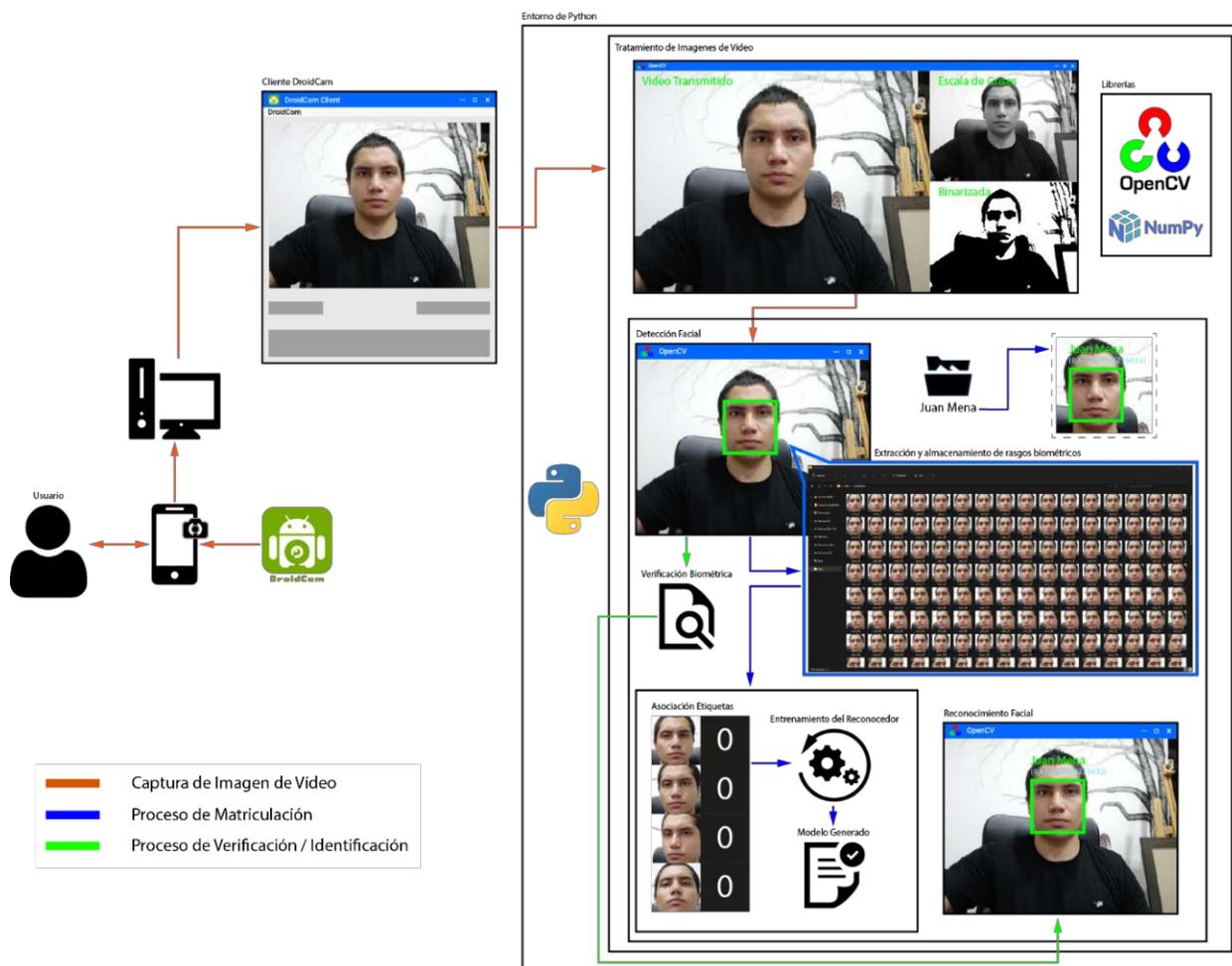


Nota. Realizado por los autores.

Solventada la necesidad de la cámara de video, se procedió con el uso de la librería de visión por computador OpenCV y NumPy, permitiendo realizar la transmisión de la imagen de vídeo a través de la ejecución de una ventana en el sistema operativo y la manipulación de las imágenes de vídeo.

Figura 84

Arquitectura descriptiva de los procesos del sistema de reconocimiento facial de la iteración 1



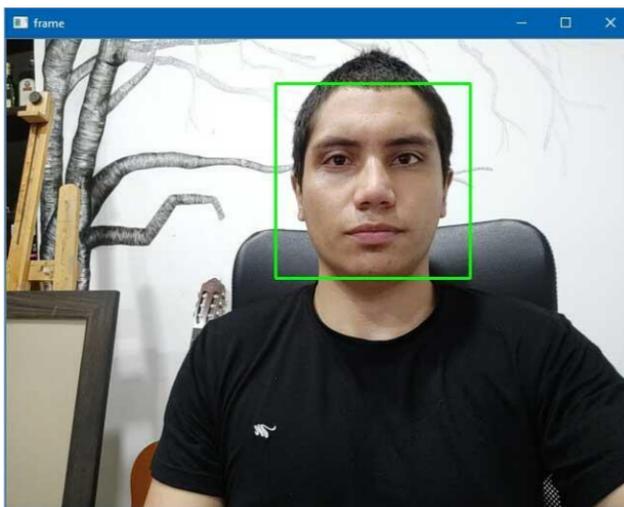
Nota. El diagrama muestra los procesos de la arquitectura del sistema de reconocimiento facial, representados por el color naranja para la captura de imagen de vídeo, el azul para el proceso de

matriculación y el verde para el proceso de verificación e identificación. Realizado por los autores.

Para la detección del rostro de la persona se hacía necesaria la conversión de la imagen de entrada en escala de grises para la implementación del clasificador de OpenCV: “Haar Cascade”.

Figura 85

Detección del rostro desde una ventana del SO ejecutando OpenCV



Nota. Realizado por los autores.

Haar Cascade fue presentado por primera vez por Viola y Jones en su presentación seminal en 2001, Haar Cascade fue un método de detección de rostros y objetos muy eficiente, ya que evitaba el consumo innecesario de recursos y permitía la detección de rostros en tiempo real. (Viola & Jones, 2001)

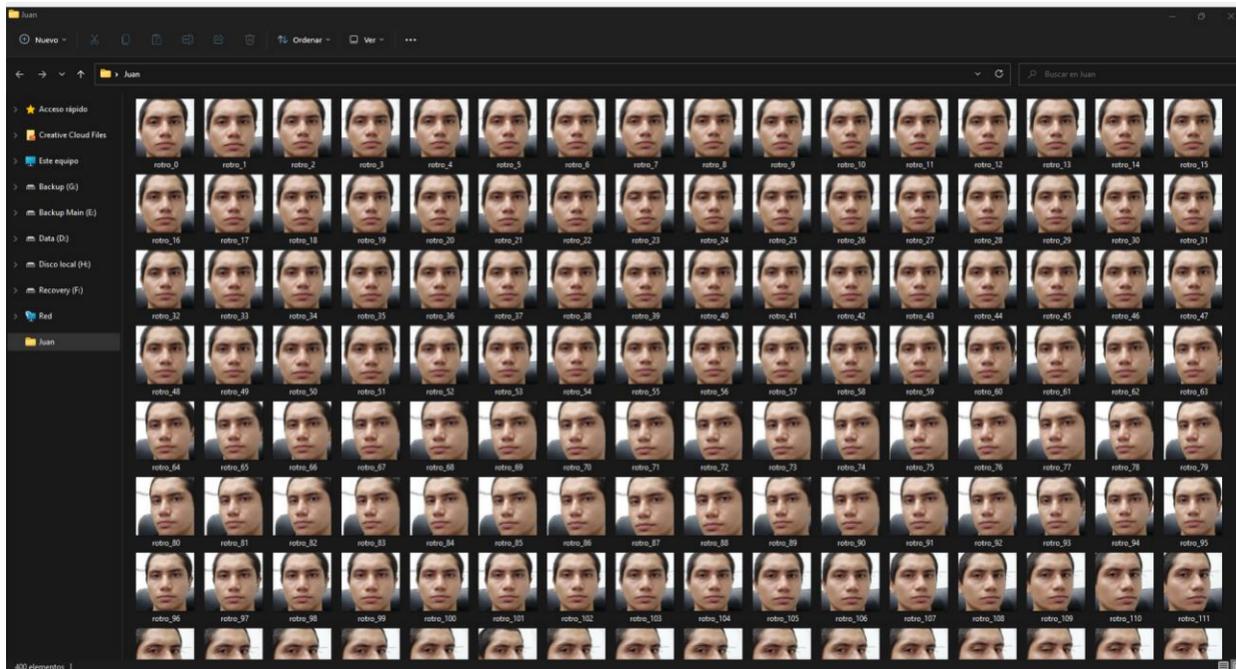
A pesar de que Haar Cascade fue presentado hace un par de décadas en el área de las ciencias de la computación, su longevidad ha permitido que en la actualidad siga jugando un papel muy importante para la construcción de clasificadores, por tal motivo, en el desarrollo del proyecto se optó por la implementación de los clasificadores de rostros frontales de Haar Cascade pre entrenados por los desarrolladores de OpenCV.

Una vez establecida la transmisión de la imagen de vídeo y la identificación del rostro en las imágenes emitidas por medio de Haar Cascade, se codificó un algoritmo que permitiera realizar la captura y el almacenamiento de los rostros en ficheros organizados con el nombre de la persona, redimensionando las fotos capturadas para estandarizar las imágenes y ejecutar un entrenamiento bajo las mismas condiciones.

Es así, como se realizó la captura de 300 imágenes del rostro de la persona, donde la persona debía gesticular varias expresiones faciales, en diferentes condiciones de luminosidad, con el fin de permitir un mayor rango de identificación al efectuar el entrenamiento del modelo.

Figura 86

Ejemplo de fichero con las fotos de los rostros capturados



Nota. Realizado por los autores.

Figura 87

Ejemplo de fotos capturas de tamaño 150 x 150



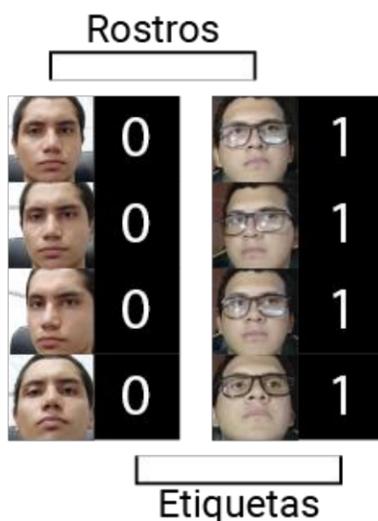
Nota. Realizado por los autores.

Una vez almacenados en ficheros organizados el conjunto de rostros de las personas detectadas, se procedió a asociar el conjunto de imágenes con una etiqueta numérica dentro de una matriz con ayuda de NumPy, buscando establecer en el modelo de entrenamiento una

relación entre el rostro y la etiqueta, de forma que cuando el modelo fuera entrenado, se pudiera lograr la identificación de la persona según el nombre del fichero.

Figura 88

Ejemplo de asociación entre el rostro y la etiqueta representativa



Nota. Realizado por los autores.

Desde este punto se experimentó con los diferentes métodos de entrenamiento: EigenFaces, FisherFaces y LBPH, con la intención de evaluar su comportamiento de identificación. Para la utilización de los métodos de entrenamiento, se requería la estandarización de las imágenes y su conversión a imágenes en blanco y negro, de acuerdo con las especificaciones estipuladas por OpenCV para el manejo de los métodos.

De esta forma se obtuvieron tres tipos de modelos diferentes en formato .xml para el reconocimiento facial, durante el entrenamiento se utilizaron los recursos del GPU ya que el

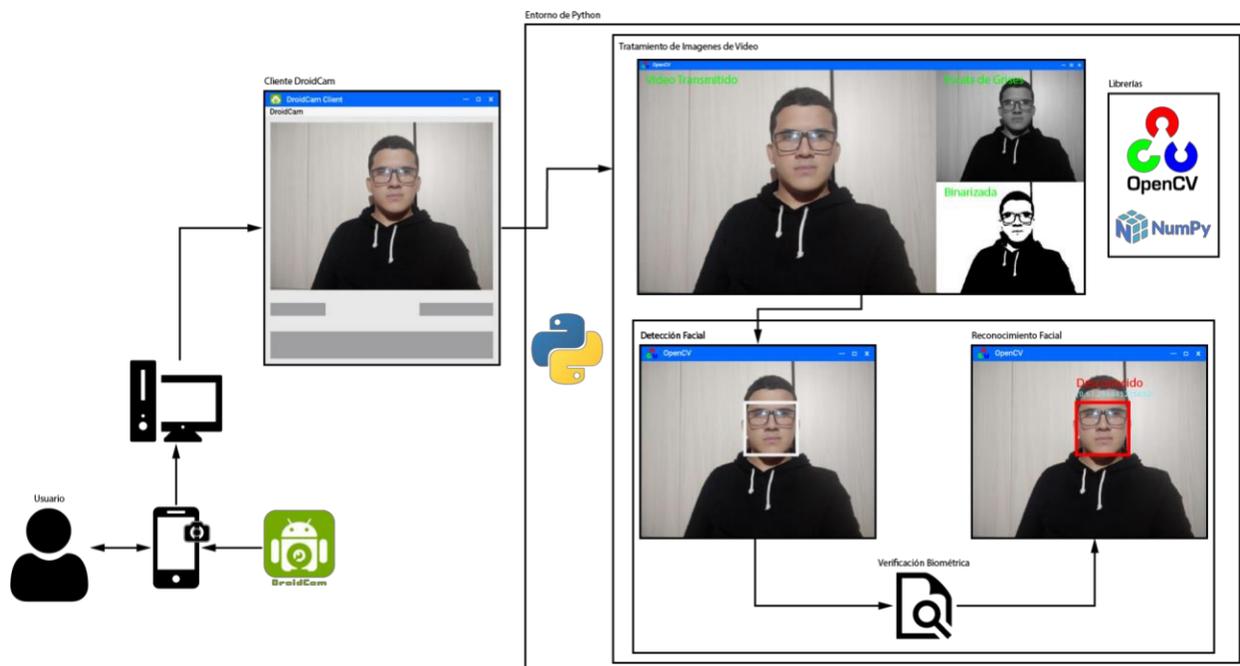
entrenamiento demanda de mucha capacidad de procesamiento en paralelo, pero durante el proceso de entrenamiento de esta iteración, la cantidad de personas de muestra para el análisis y evaluación del comportamiento del algoritmo, no era tan demandante, por consiguiente el proceso de entrenamiento consumió menos recursos y fue más eficiente, con tiempos de entrenamiento reducidos.

El fundamento de generar como resultado un modelo en .xml se determinó por la complejidad de procesamiento del entrenamiento del modelo, ya que si no se hiciera de esta forma generaría una deficiencia en el reconocimiento facial, debido a que el sistema debería realizar el proceso de entrenamiento cada vez que se requiriera reconocer una persona, en cambio, al generar un modelo pre entrenado, el sistema solo debería acceder al recurso generado, estableciendo un punto de partida para el reconocimiento de las personas.

Una vez generado el modelo, si la “cámara de videovigilancia” detectaba alguna persona que no estuviera registrada en el sistema, el algoritmo posicionaría sobre el rostro de la persona un recuadro rojo y la palabra desconocido, alertando sobre el ingreso de un posible intruso en el sistema, aunque, una vez realizados todos los procesos del reconocimiento facial, el usuario podría quedar registrado en el sistema, siendo reconocido como un usuario más dentro de este.

Figura 89

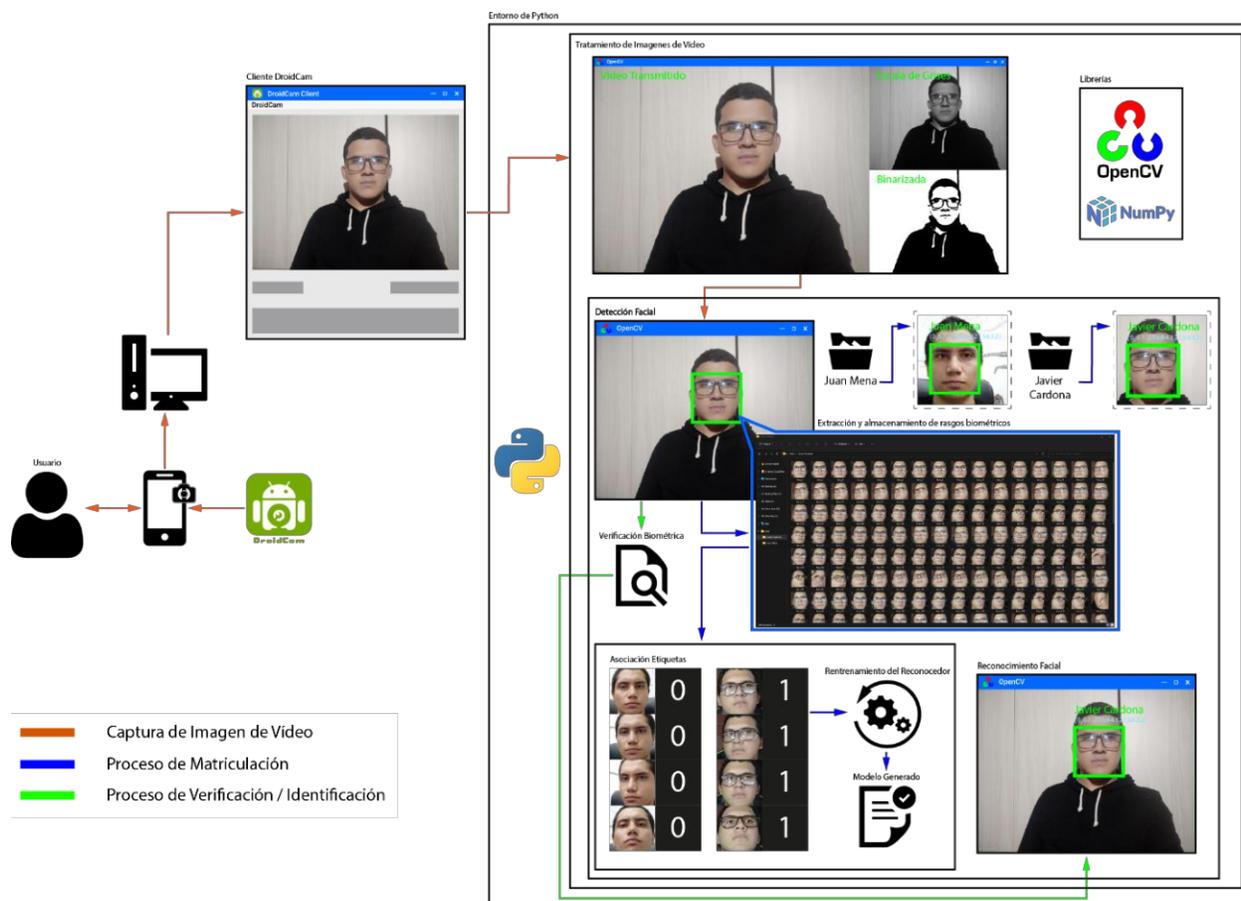
Proceso de detección de personas desconocidas de la iteración 1



Nota. En el diagrama se muestran las diferentes fases de captura, detección y reconocimiento de personas desconocidas dentro del sistema. Realizado por los autores.

Figura 90

Matriculación, verificación e identificación de un nuevo usuario en la iteración 1

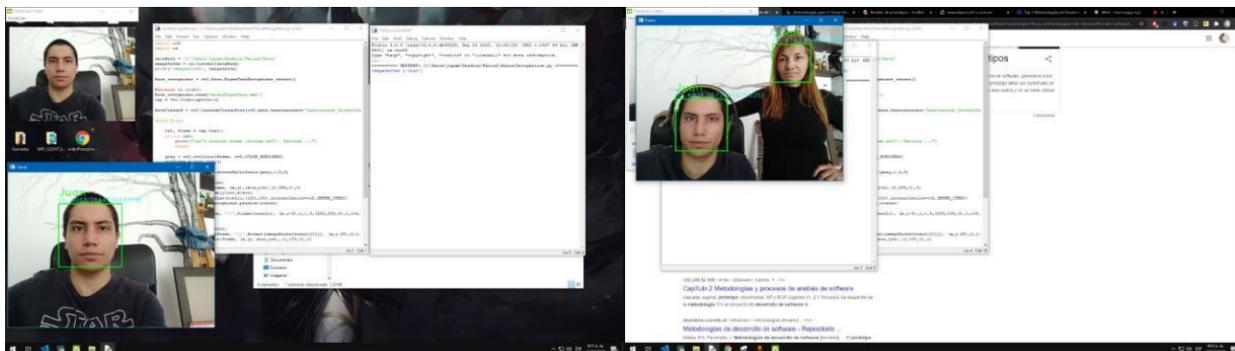


Nota. El diagrama muestra la aplicación de todos los procesos del sistema de reconocimiento facial para la inscripción de un usuario que antes era desconocido. Realizado por los autores.

De esta forma se logró verificar la capacidad de OpenCV para permitir la detección y el reconocimiento facial de distintas personas, validando el resultado del modelo entrenado según el rasgo biométrico de cada persona dentro del mismo fotograma.

Figura 91

Ejemplo de reconocimiento facial de dos personas con OpenCV



Nota. La figura muestra el despliegue de las ventanas en el sistema operativo y la capacidad de OpenCV para reconocer múltiples personas en una sola imagen de vídeo. Realizado por los autores.

5.2.1.3 Evaluación del prototipo

En este punto, el prototipo fue netamente experimental, se logró: transmitir la imagen en una ventana del sistema operativo, detectar el rostro en la imagen de vídeo, capturar la imagen detectada, organizar ficheros de acuerdo a la persona que fue capturada, asociar con una etiqueta el fichero con la persona, generar los modelos de entrenamiento de acuerdo al método escogido y realizar el reconocimiento facial de acuerdo con las personas almacenadas en el conjunto de datos.

Es así como en la primera iteración se cumplió con varios de los requerimientos planteados inicialmente, aunque muchos se cumplieron, algunos se efectuaron en término medio y otros no se lograron, debido a problemas de infraestructura o que debían satisfacerse a lo largo de las demás iteraciones.

5.2.1.3.1 Nivel de satisfacción de los requerimientos. Con el fin de retroalimentar el sistema y verificar la necesidad de realizar modificaciones a los requerimientos estipulados en primera instancia, se llevó a cabo un estudio de satisfacción de cada uno de los requerimientos de acuerdo con el resultado de la iteración y el comportamiento del sistema en general, planteando la siguiente nomenclatura para evaluar el nivel de satisfacción de los requerimientos:

- **S:** Satisfecho
- **MS:** Medianamente Satisfecho
- **IS:** Insatisfecho

Requerimientos iniciales

Tabla 35

Nivel de satisfacción de los requerimientos iniciales - Iteración 1

Nivel de Satisfacción de los Requerimientos Iniciales		
No.	Nivel de Satisfacción	Observación
RIS1	S	El sistema, aunque no era un sistema netamente estructurado, permitía la detección de rostros en tiempo real.
RIS2	S	El sistema permitía el almacenamiento de los rostros en ficheros organizados con 300 imágenes redimensionadas del rostro de la persona, según le correspondía.
RIS3	MS	No sé contaba con una cámara de videovigilancia, por tal motivo se utilizó DroidCam para solventar dicha necesidad.
RIS4	S	Se realizó la renderización del a imagen de vídeo por medio de la GPU, a través de una ventana del sistema operativo.

Nivel de Satisfacción de los Requerimientos Iniciales		
No.	Nivel de Satisfacción	Observación
RIS5	S	El sistema estuvo conectado las 24 horas del día a la red eléctrica.
RIS6	MS	El sistema permitía reconocer y almacenar personas identificadas en conjuntos de datos, pero no se había estructurado una base de datos.
RIS7	MS	El sistema permitía reconocer personas no identificadas, pero no se había estructurado una base de datos.
RIS8	IS	El prototipo no se había trabajado en el Edificio de Ingeniería.
RIS9	IS	No se realizó la obtención del vídeo por medio de la cámara de videovigilancia, ni un circuito cerrado de televisión.
RIS10	S	El sistema permitía permanentemente el procesamiento de las imágenes obtenidas por la cámara.
RIS11	IS	El sistema aun no podía ser implementado.
RIS12	IS	El sistema aun no podía ser implementado, pero se tuvo a consideración la reducción de agentes externos.
RIS13	IS	No se contaba con cámara de videovigilancia, pero se tuvo a consideración la ubicación idónea para la captura de las personas, para el proceso de identificación.

Nota. Realizado por los autores.

Requerimientos funcionales

Tabla 36

Nivel de satisfacción de los requerimientos funcionales - Iteración 1

Nivel de Satisfacción de los Requerimientos Funcionales		
No.	Nivel de Satisfacción	Observación
RU1	IS	

Nivel de Satisfacción de los Requerimientos Funcionales		
No.	Nivel de Satisfacción	Observación
		El sistema no se había implementado en el Edificio de Ingeniería.
RU2	S	El sistema permitía la adquisición de la información para realizar el entrenamiento.
RU3	S	El sistema permitía el almacenamiento de los rostros de las personas en directorios estructurados y organizados, para la realización del entrenamiento del modelo.
RU4	IS	No se había planteado el CRUD, para la gestión del sistema.
RU5	IS	No se había planteado el CRUD para la gestión del sistema.
RU6	IS	No se había planteado el CRUD para la gestión del sistema.
RU7	IS	No se había planteado el CRUD para la gestión del sistema.
RU8	IS	No se había planteado el CRUD, para la gestión del sistema.
RU9	IS	No se había planteado el CRUD, para la gestión del sistema.
RU10	IS	No se habían planteado perfiles de administración.
RU11	IS	El sistema no contaba con las condiciones para permitir identificar rostros sin ser supervisado.

Nota. Realizado por los autores.

Requerimientos no funcionales

Tabla 37*Nivel de satisfacción de los requerimientos no funcionales - Iteración 1*

Nivel de Satisfacción de los Requerimientos No Funcionales		
No.	Nivel de Satisfacción	Observación
RA1	IS	El sistema no podía ser implementado en el Edificio de Ingeniería.
RA2	MS	A pesar de que el sistema no podía ser implementado, se manejó mediante una red de área local.
RA3	MS	El sistema se estructuró mediante un servidor local, con ventilación idónea.
RA4	S	El sistema se desarrolló mediante un CPU capaz de realizar cualquier tipo de procesamiento complejo.
RA5	S	El sistema contaba con una GPU para el tratamiento y renderizado de imágenes en tiempo real.
RA6	S	El sistema contaba con un PSU adecuada, para brindar el suministro de energía suficiente a todos los componentes del servidor.
RA7	S	El sistema contaba con una PSU con certificación, para garantizar la estabilidad del suministro de energía.
RA8	S	El sistema contaba con una tarjeta madre con conexión a Ethernet.
RA9	S	El sistema contaba con una GPU con características adecuadas para realizar un entrenamiento efectivo de los modelos.
RA10	S	El sistema contaba con una GPU con características adecuadas para solventar cualquier necesidad de procesamiento en paralelo.
RA11	MS	A pesar de que no se contaba con una cámara de videovigilancia, se realizó el análisis de los requerimientos necesarios para la cámara de videovigilancia y se realizó su selección, con base en términos de calidad precio.

RA12	S	Las imágenes eran procesadas eficientemente por el CPU.
RA13	S	El sistema contaba con almacenamiento suficiente para almacenar volúmenes considerables de información.
RA14	S	El CPU era compatible con cualquier tipo de arquitectura de SO disponibles.
RA15	S	El CPU era de última generación.
RA16	S	El CPU ofrecía una frecuencia de reloj base de 3.8Mhz y podía ser overclockeado hasta una frecuencia de 4.6Mhz.
RA17	IS	A pesar de que el sistema poseía GPU dedicada, el CPU no contaba con GPU integrada.
RA18	IS	No se contaba con una cámara de videovigilancia.
RA19	S	El sistema se desarrolló en Windows 10, pero podía ser implementado en cualquier SO, además, se desarrolló con herramientas de desarrollo de código abierto como Visual Studio Code, PyCharm, entre otras.
RA20	S	Se implementó el lenguaje de programación Python, por su nivel de compatibilidad con OpenCV y su capacidad de manejar tecnologías de inteligencia artificial de manera práctica.
RA21	S	El sistema permitía la ejecución de visión artificial en tiempo real.
RA22	S	El sistema permitía la ejecución de multiprocesos de forma eficaz.
RA23	S	El sistema permitía la utilización de los recursos de la GPU, para el entrenamiento de los modelos.
RA24	IS	No se había estipulado la base de datos.

RA25	S	Las herramientas de software eran compatibles entre sí para el uso de las bibliotecas de aprendizaje automático.
RA26	S	Las herramientas de software eran compatibles con tecnología de cómputo en paralelo.
RA27	S	El sistema permanecía conectado a la red eléctrica.

Nota. Realizado por los autores.

5.2.1.4 Refinamiento del prototipo

Varios de los requerimientos fueron satisfechos en la primera iteración, pero de acuerdo con los inconvenientes presentados, fue necesario realizar una nueva proyección del prototipo reestructurando las herramientas de desarrollo y modificando la arquitectura del sistema con el fin de orientar el desarrollo del proyecto hacia la web.

Por tal motivo, para los requerimientos de arquitectura de la segunda iteración, se planteó una investigación de los marcos de trabajo que permitieran el desarrollo del prototipo en entorno Python orientado a la web.

5.2.2 Iteración 2

5.2.2.1 Requerimientos

Durante el desarrollo de la segunda iteración del proyecto, se llevó a cabo inicialmente un proceso de refinamiento de requerimientos, de acuerdo con el análisis y exclusión de varias

herramientas que habían sido consideradas en la primera iteración. Por lo tanto, se decidieron explorar marcos de trabajo como Flask y Django, marcos que, aunque permitían la creación de aplicaciones web de forma sencilla en Python, fueron descartados por los desarrolladores debido a la practicidad de utilizar marcos de trabajo propiamente del desarrollo web como: Angular, ya que “Angular es un marco de diseño de aplicaciones y una plataforma de desarrollo para crear aplicaciones de una sola página eficientes y sofisticadas”. (Google, 2021)

De esta forma, se anexó un nuevo requerimiento dentro de los requerimientos no funcionales, específicamente los requerimientos de software, estableciéndose como: **RA28**.

Tabla 38

Anexo de requerimiento no funcional - Iteración 2

No.	Requerimientos No Funcionales	Nivel de Prioridad		
		Alta	Media	Baja
Requerimientos de Software				
RA28	Se requiere de software orientado a la web que permita la visualización de las interfaces de usuario.	x		

Nota. Realizado por los autores.

5.2.2.2 Diseño y construcción

5.2.2.2.1 Hardware. Dentro de esta iteración el hardware no presentó cambios, debido a esto, se utilizó el mismo que fue planteado en la primera iteración.

5.2.2.2.2 Software. Dentro de las herramientas implementadas, se siguió manejando Python como lenguaje principal, debido a que su tipado dinámico, su naturaleza interpretada y su alta compatibilidad con bibliotecas de procesamiento de visión por computadora y aprendizaje profundo, lo convierten en el lenguaje ideal para el desarrollo aplicaciones multiplataforma de inteligencia artificial. (Serrano, Universidad Don Bosco, 2019)

Así mismo, se estableció MySQL como gestor de base de datos, partiendo del hecho de que es un gestor relacional de código abierto que cuenta con altos estándares de seguridad (Mauricio, 2020), compatibilidad en importantes plataformas computacionales como macOS, Linux, Ubuntu y Microsoft Windows, (Ipanaqué, 2020) y con niveles nulos de restricciones en cuanto al desarrollo de aplicaciones se refiere. (Noguera, 2012)

Del mismo modo, se buscaron herramientas como UVICORN para realizar la implementación de WebSockets, debido a que UVICORN es una implementación de servidor ASGI ultrarrápida (Uvicorn, 2021), que permite el lanzamiento de un servicio asíncrono de forma sencilla pero robusta; permitiendo establecer una arquitectura donde UVICORN actuaría como servidor y Angular como cliente.

5.2.2.2.3 Arquitectura. Una vez estipuladas las herramientas para su utilización en la segunda iteración, se decidió realizar un proceso de experimentación por medio de la transmisión de imágenes de vídeo a través del uso de librerías para el manejo de WebSockets, permitiendo realizar transmisiones en “tiempo real” que afianzaran el proceso de reestructuración de la arquitectura inicial.

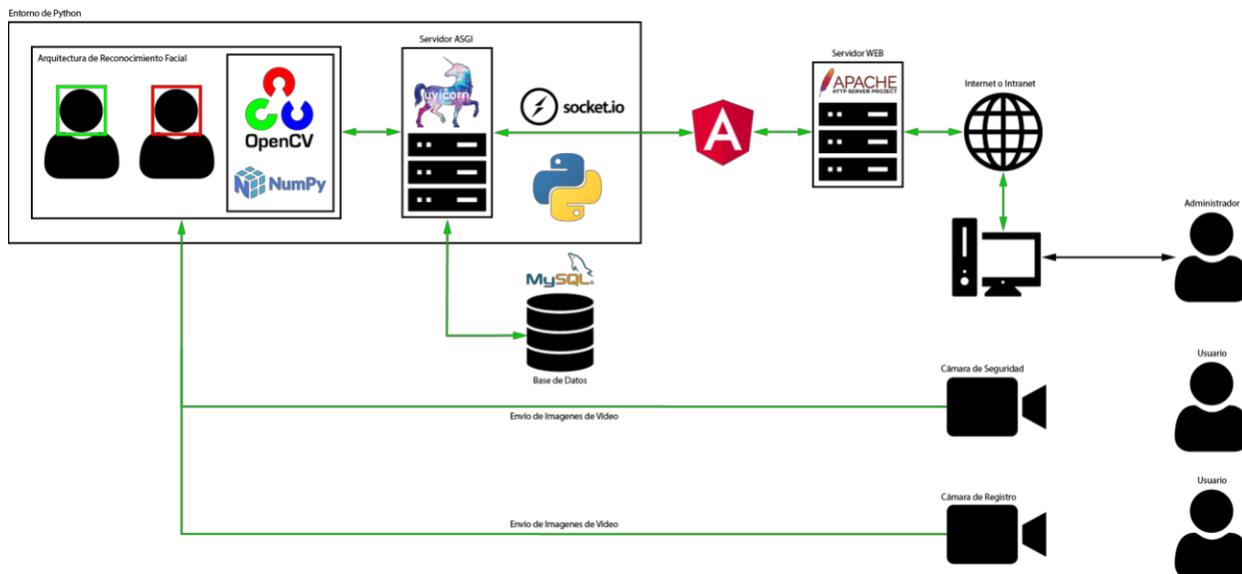
“El protocolo WebSockets permite la comunicación bidireccional entre un cliente que ejecuta código que no es de confianza en un entorno controlado y un host remoto que ha optado por recibir comunicaciones desde ese código”. (A. Melnikov, 2011)

“El objetivo de esta tecnología es proporcionar un mecanismo para aplicaciones basadas en navegación que necesitan comunicación bidireccional con servidores que no depende de la apertura de múltiples conexiones HTTP (por ejemplo, usando XMLHttpRequest o <iframe>s y sondeos largos)”. (A. Melnikov, 2011)

La arquitectura planteada en la segunda iteración, se estructuró de la siguiente manera:

Figura 92

Arquitectura estructural e integral de las tecnologías utilizadas en la iteración 2



Nota. En el diagrama se logra visualizar las interacciones entre las diferentes tecnologías y las partes que integran el sistema, es decir: El sistema de reconocimiento facial, los usuarios, la base de datos, los servicios y el cliente web. Realizado por los autores.

Dentro de la arquitectura se manejaron los mismos algoritmos plateados en la primera iteración para la transmisión de la cámara de vídeo, la detección del rostro, el almacenamiento de rostros, el entrenamiento del modelo y el reconocimiento facial, pero las imágenes dejaron de ser presentadas a través del sistema operativo.

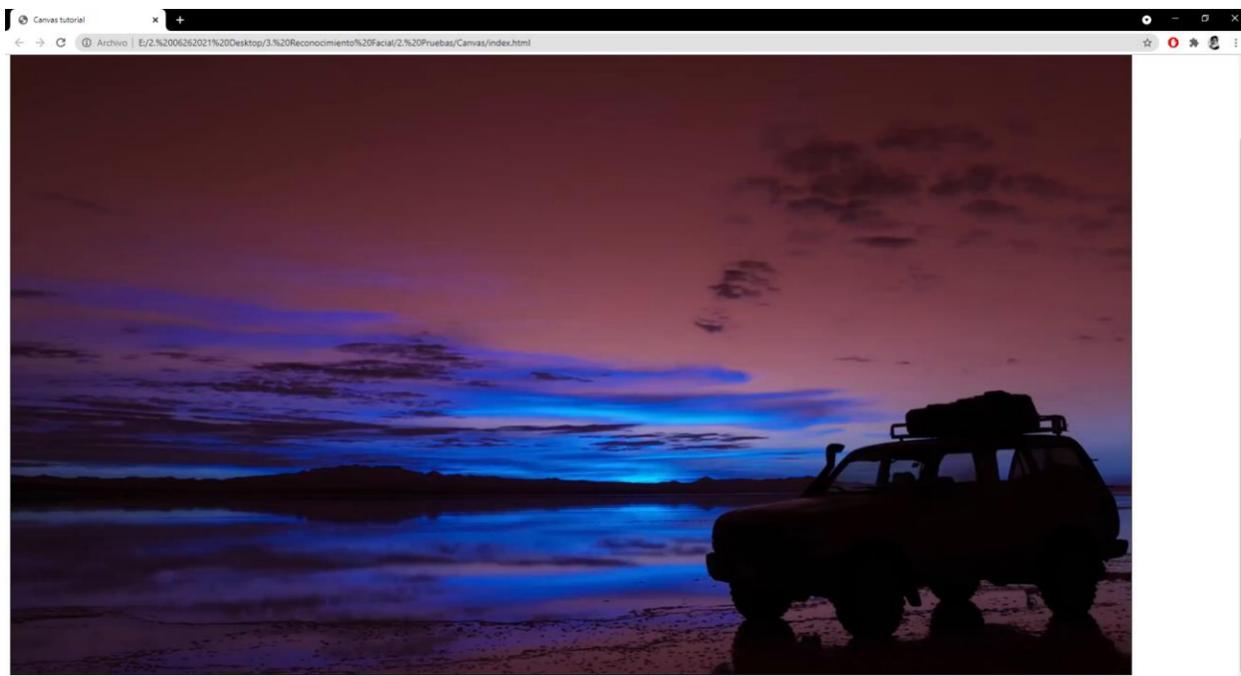
En este punto del desarrollo se presentaron varios inconvenientes, debido a que se desconocía la forma en que debían ser tratadas las imágenes capturadas para ser enviadas a través de los WebSockets, por este motivo, a partir del proceso investigativo se determinó que la forma más práctica para la transmisión de las imágenes, debía efectuarse mediante la codificación en formato Base64, debido a que los navegadores están diseñados para entender caracteres ASCII.

En un principio se planteó realizar la transmisión de la imagen de vídeo utilizando un arreglo matricial sin codificación en Base64, transfiriendo la imagen desde el servicio de UVICORN hacía el navegador e interpolándola dentro de la etiqueta “canvas” de HTML por medio del lenguaje de programación TypeScript utilizado en las aplicaciones Angular.

De esta forma se logró proyectar una imagen en el navegador, capturada desde Python por medio de OpenCV y transmitida a través de los WebSockets con ayuda de UVICORN, aunque dicho método de transferencia fue poco eficiente, debido a que las imágenes según su resolución poseían un conjunto de datos matriciales muy demandante y complejo para ser transferido en tiempo real hacía el navegador.

Figura 93

Prueba de renderizado de una imagen en la etiqueta canvas, enviada a través de la implementación de WebSockets

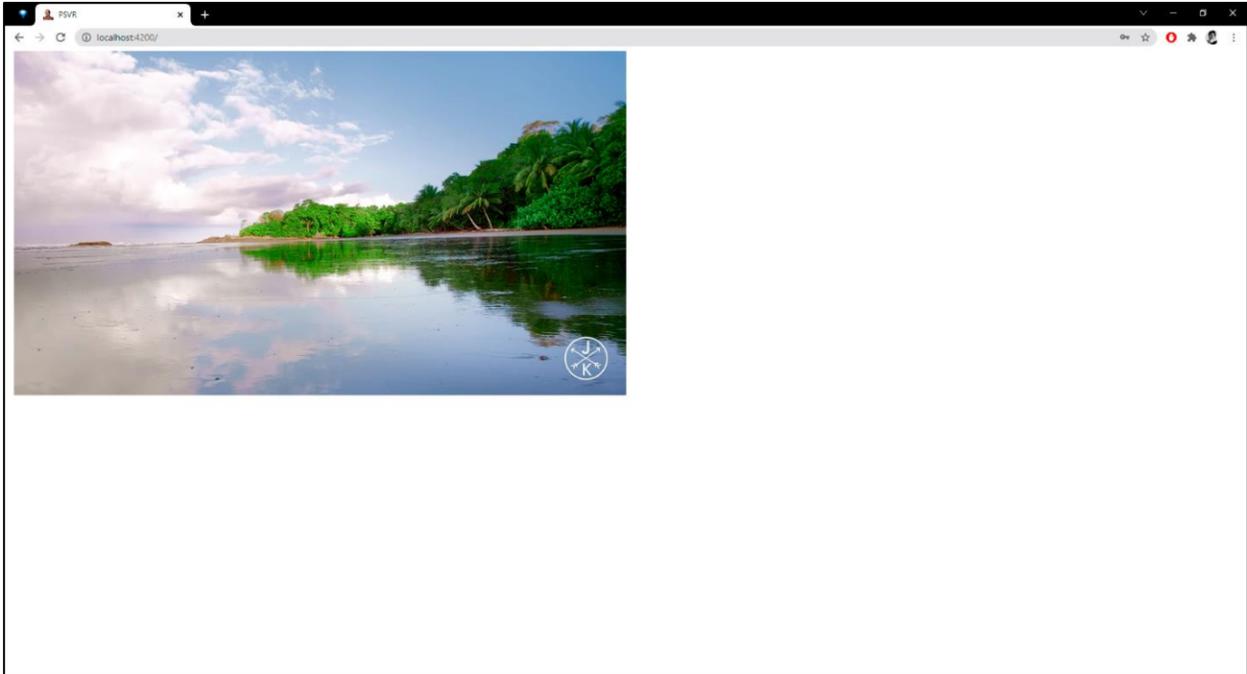


Nota. Realizado por los autores.

Para solucionar el problema se elaboraron estructuras de software que permitieran la codificación en Base64 de los fotogramas de un vídeo experimental, permitiendo ser renderizadas e interpolados en la etiqueta “img” en HTML, optimizando el proceso de transferencia de datos entre el servidor y el navegador. En este punto se presentaron inconvenientes debido a la resolución del vídeo, ya que demandaba alto procesamiento de información, por dicha razón, se buscaron soluciones para disminuir el tamaño de los fotogramas del vídeo, ayudando a la reducción de latencias en la transmisión de la imagen.

Figura 94

Vídeo transmitido a través del navegador por medio de WebSockets

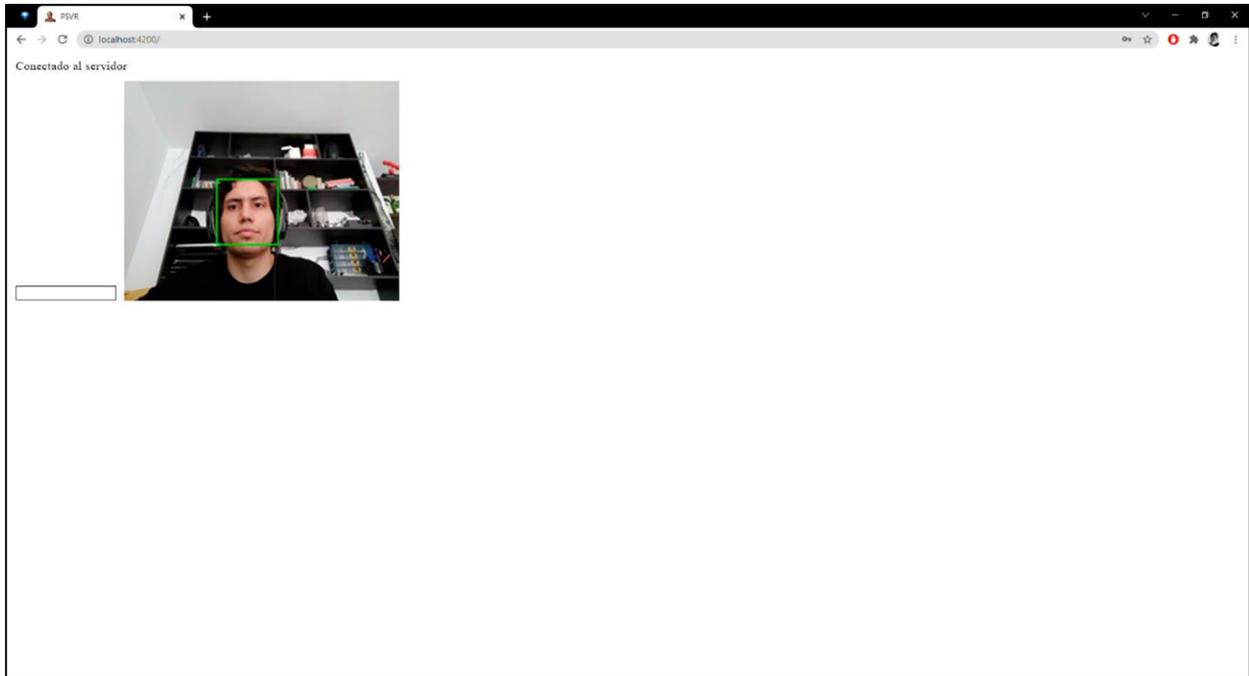


Nota. Realizado por los autores.

Una vez conseguida la optimización de vídeo y la ejecución de pruebas por medio del vídeo experimental, se realizó un programa para ejecutar la transferencia de la imagen de vídeo capturada directamente de la cámara, enviada a través de los WebSockets y visualizada en Angular.

Figura 95

Experimentación de captura y presentación de imagen de vídeo por medio de DroidCam a través del navegador, utilizando UVICORN para la gestión de websockets y Angular



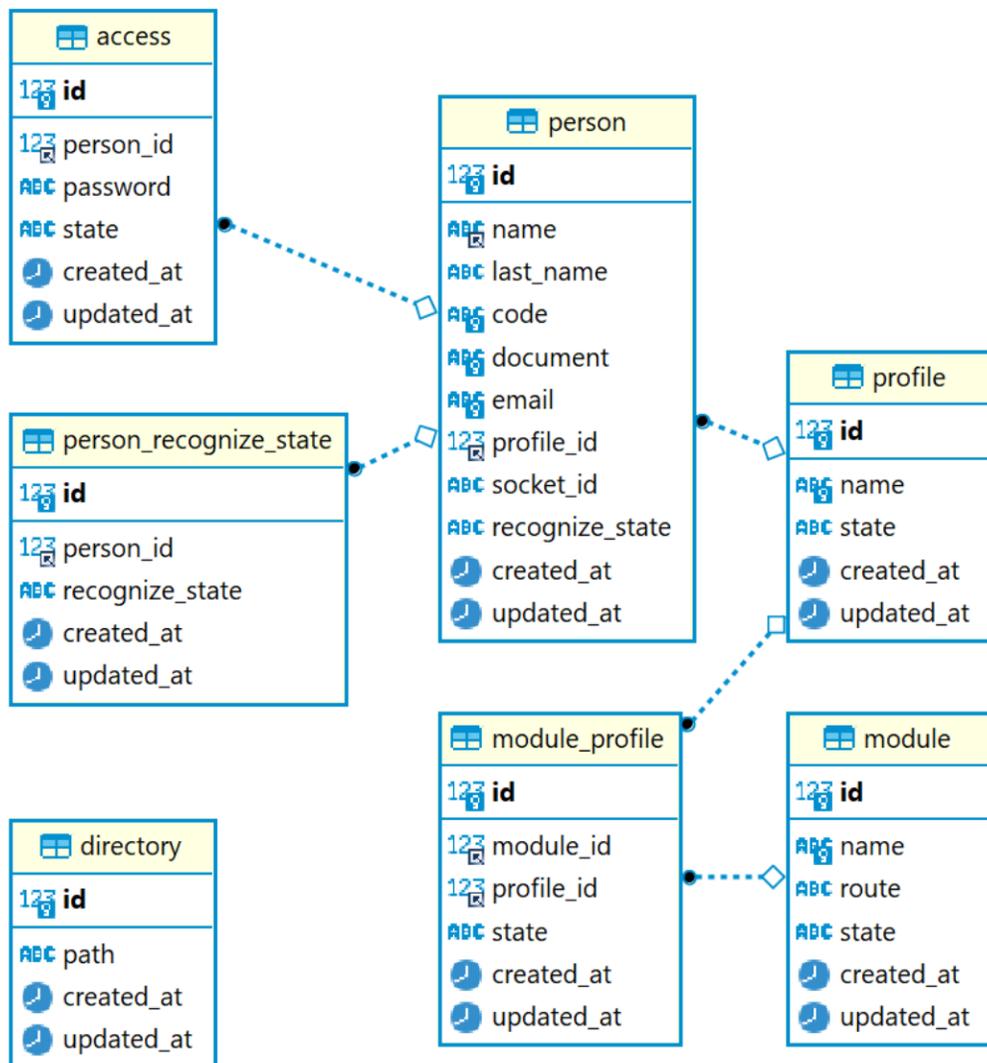
Nota. Realizado por los autores.

En este punto del proyecto, se procedió con la construcción de la arquitectura de software para la gestión y almacenamiento de la información de acuerdo con los requerimientos estipulados dentro la primera iteración, planteando una primera versión del modelo relacional del sistema.

5.2.2.2.4 Modelo relacional del sistema.

Figura 96

Modelo relacional del sistema en la iteración 2.



Nota. Realizado por los autores.

5.2.2.2.5 Detalle de las tablas del modelo relacional.

Tabla 39*Detalle de las tablas del modelo relacional - Iteración 2*

Detalle de las Tablas del Modelo Relacional		
Nombre de la Tabla	Descripción	Relación
“access”	Permite gestionar los accesos de los usuarios dentro del sistema, almacenando un identificador, la contraseña, el estado del acceso y la fecha de creación del registro.	Uno a muchos con la tabla “person”
“person”	Permite gestionar la información de las personas registradas dentro del sistema, almacenando un identificador, el nombre, el apellido, el código universitario, el documento de identificación, el correo electrónico, el identificador del socket, el estado de reconocimiento, la asociación con la tabla “profile” y “person_recognize_profile” y la fecha de creación del registro.	Muchos a uno con la tabla “person_recognize_state” Uno a muchos con la tabla “profile”
“person_recognize_state”	Permite almacenar los diferentes estados de reconocimiento de los usuarios del sistema, almacenando un identificador, el estado de reconocimiento, la asociación con la tabla “person” y la fecha de creación del registro.	Uno a muchos con la tabla “person”
“profile”	Cumple la función de ser una tabla intermedia entre la tabla “person” y “module_profile”, con la finalidad de establecer un registro de relación, almacenando un identificador, el nombre del perfil, el estado de actividad y la fecha de creación del registro.	Muchos a uno con la tabla “person” Muchos a uno con la tabla “module_profile”
“module_profile”	Permite gestionar el control de estados de los módulos relacionados	Uno a muchos con la tabla “profile”

Detalle de las Tablas del Modelo Relacional		
Nombre de la Tabla	Descripción	Relación
	de acuerdo con el perfil relacionado, almacenando un identificador, el identificador del módulo, el identificador del perfil, el estado de actividad y la fecha de creación del registro.	Uno a muchos con la tabla “module”
“module”	Permite gestionar la información de los módulos, almacenando un identificador, el nombre del módulo, la ruta específica para Angular, el estado de actividad y la fecha de creación del registro.	Muchos a uno con la tabla “module_profile”
“directory”	Permite gestionar la información del directorio temporal, almacenando un identificador, la ruta y la fecha de creación del registro.	No posee relación.

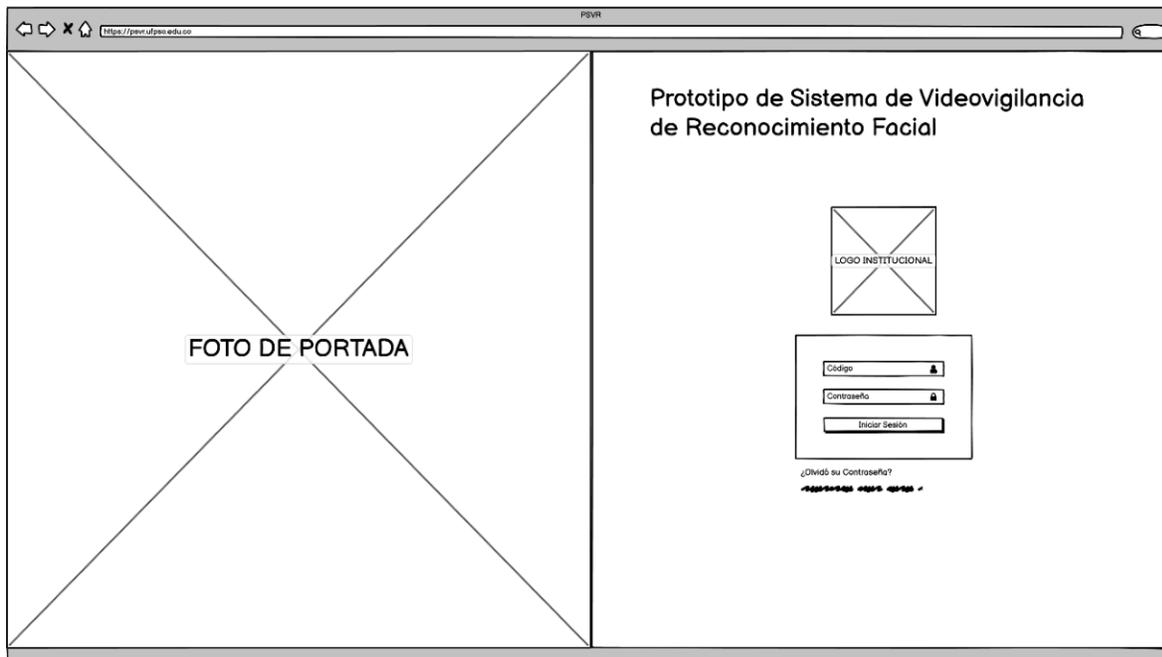
Nota. Realizado por los autores.

5.2.2.2.6 Creación de los mockups. Dentro de la segunda iteración se plantearon diferentes mockups para la estructuración de los componentes de interfaz de usuario y usabilidad del sistema, siendo estos:

Interfaz de presentación y acceso al sistema. En la interfaz de presentación y acceso al sistema se planteó un diseño minimalista, donde se visualizaría una imagen de portada relacionada al sistema o de la institución educativa, el nombre del sistema, el logo institucional, el formulario de acceso para el ingreso y la opción de restablecimiento de contraseña si se diera el caso de haberla olvidado.

Figura 97

Mockup de la interfaz de presentación y acceso al sistema

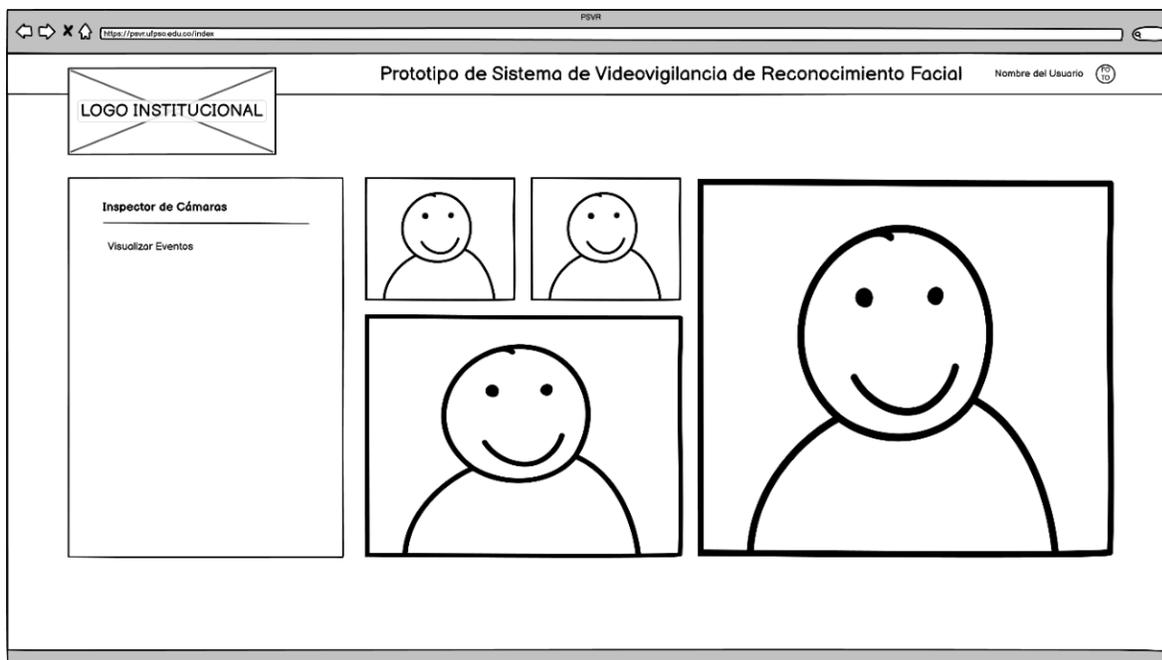


Nota. Realizado por los autores.

Interfaz de monitoreo. Para la interfaz de monitoreo se diseñó una estructura que permitiera a los usuarios con acceso al sistema, visualizar en tiempo real el monitor con las diferentes cámaras de videovigilancia asociadas, controlando los eventos captados por estas. De igual forma, se planteó visualizar el nombre del sistema, el logo institucional, el menú de administración de los diferentes módulos y las opciones de usuario para cerrar sesión.

Figura 98

Mockup de la interfaz de monitoreo

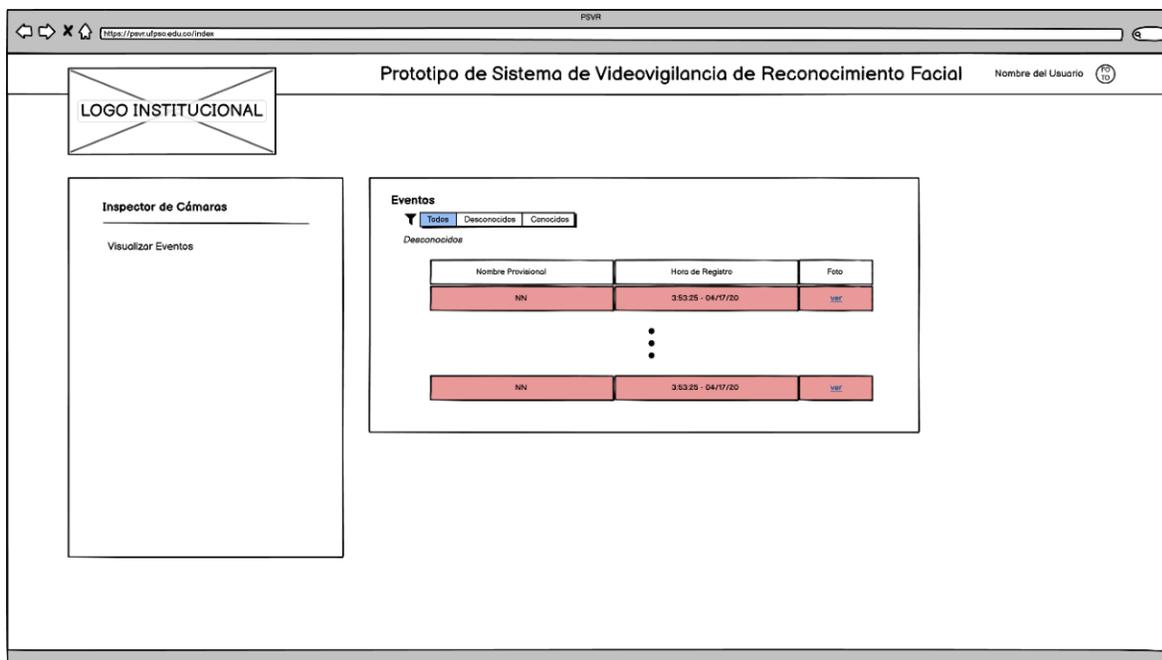


Nota. Realizado por los autores.

Interfaz de inspección de personas desconocidas. Dentro de las opciones posibles para el control de eventos del sistema, se planteó que los usuarios con capacidades administrativas, tuvieran la posibilidad de filtrar las detecciones realizadas por la cámara, como la opción de verificar todos los registros de las personas desconocidas y conocidas o de los registros independientes según su categoría.

Figura 99

Mockup de la interfaz de inspección de personas desconocidas



Nota. Realizado por los autores.

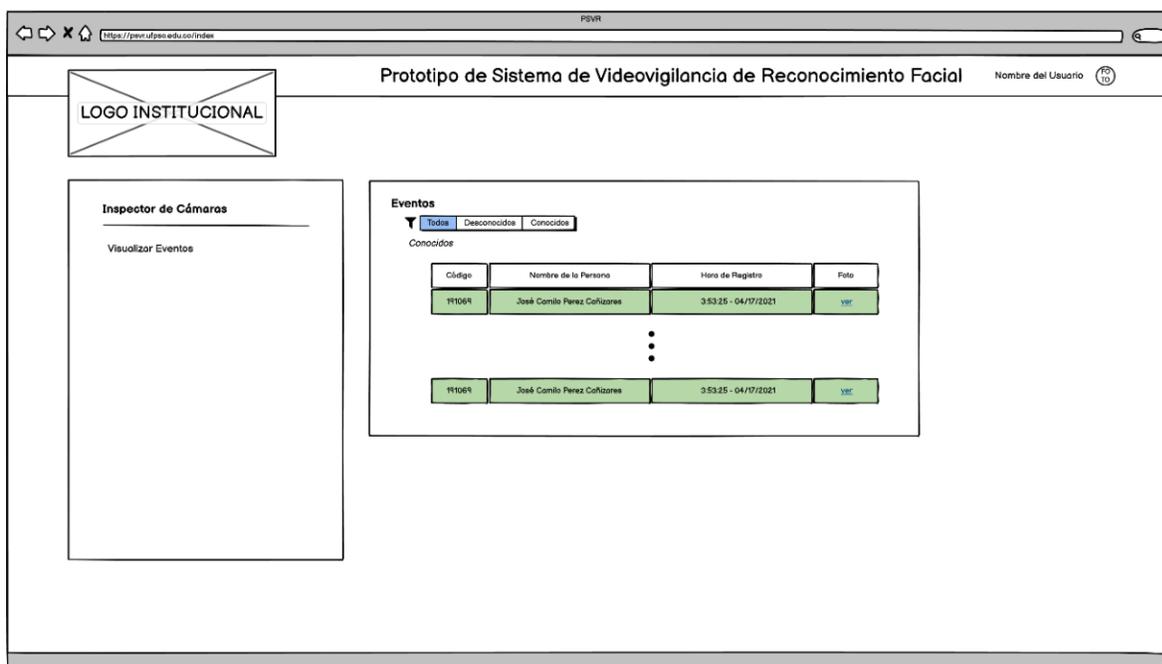
Quando el usuario filtrara por la opción de desconocidos, podría visualizar una tabla con los datos de:

- **Nombre Provisional:** Nombre de registro para usuarios no identificados con una nomenclatura específica.
- **Hora de Registro:** Hora y fecha en la que se realizó la identificación de la persona no identificada dentro del sistema.
- **Foto:** Opción para desplegar una ventana emergente para revisar la evidencia fotográfica de la persona desconocida.

Interfaz de inspección de personas conocidas.

Figura 100

Mockup de la interfaz de inspección de personas conocidas



Nota. Realizado por los autores.

Para el caso de las personas conocidas dentro del sistema, se podría visualizar una tabla con los siguientes datos:

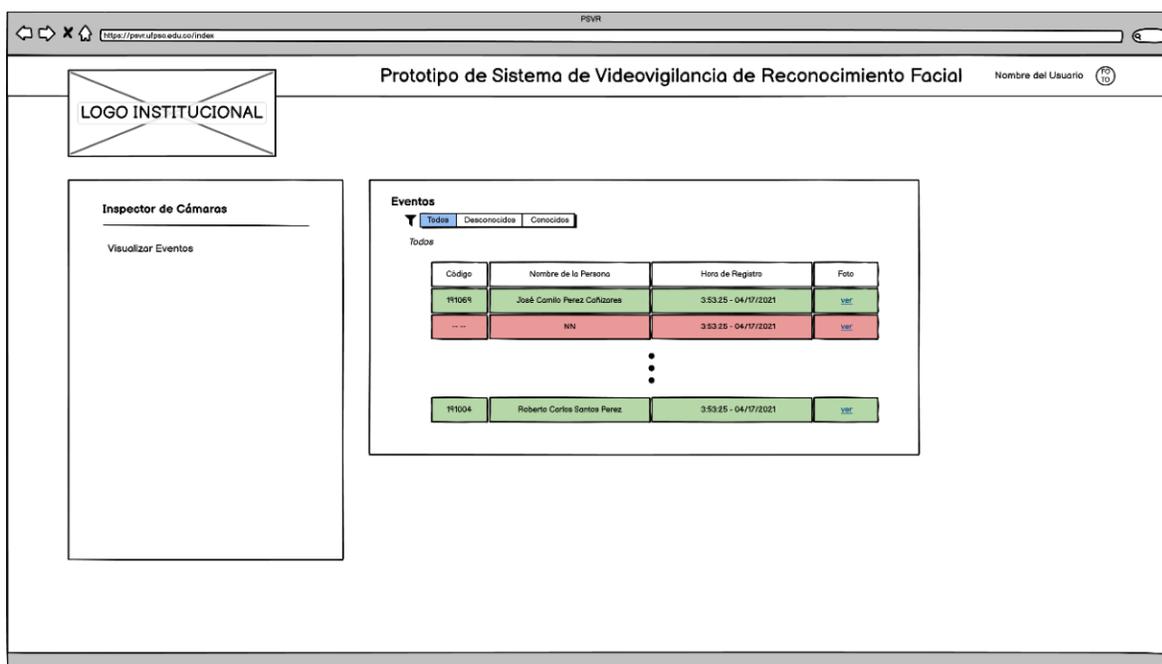
- **Código:** Código de la persona asociado según su perfil dentro de la institución.
- **Nombre de la Persona:** Nombre completo asociado al código del perfil.
- **Hora de Registro:** Hora y fecha en la que se realizó la identificación de la persona identificada dentro del sistema.

- **Foto:** Opción para desplegar una ventana emergente para revisar la evidencia fotográfica de la persona conocida.

Interfaz de inspección de personas conocidas y desconocidas. La interfaz de inspección de personas conocidas y desconocidas, se estructuró con base en la combinación de ambos registros (conocidos o desconocidos), según el orden en que iban siendo registrados en el sistema.

Figura 101

Mockup de la interfaz de inspección de personas conocidas y desconocidas

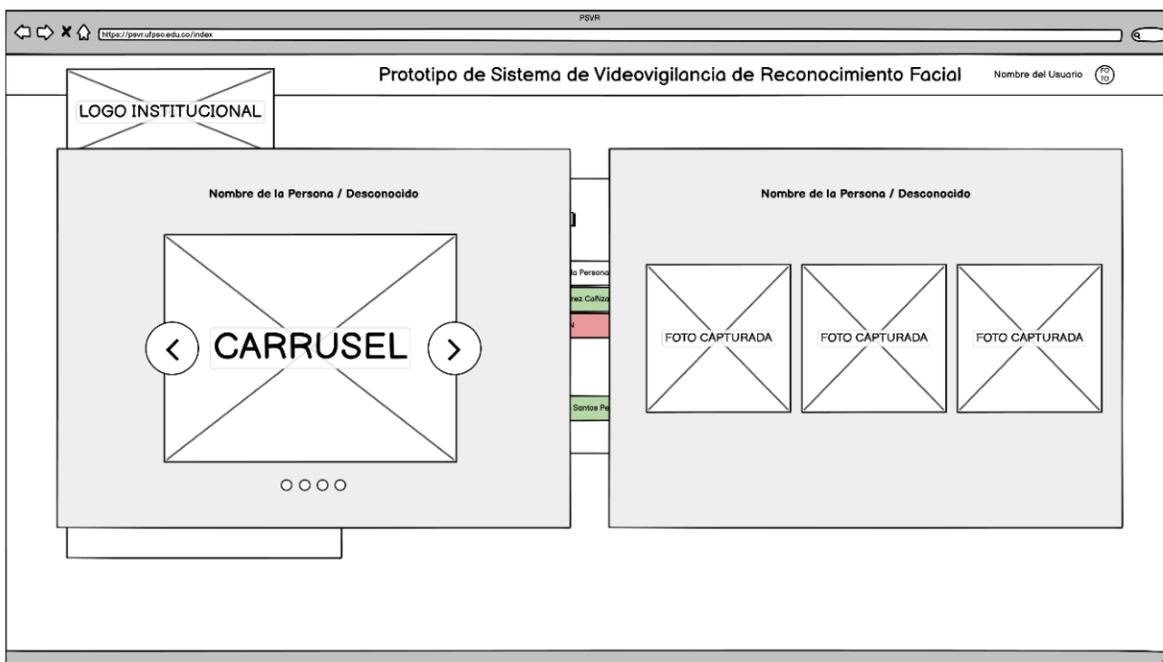


Nota. Realizado por los autores.

Interfaz detalle de personas conocidas o desconocidas. La interfaz de detalle de personas conocidas o desconocidas representaba la posibilidad de presentar mediante una ventana emergente la evidencia fotográfica captada por la cámara de videovigilancia, para que los usuarios que monitoreaban el sistema, pudieran validar determinada información según lo requerían.

Figura 102

Mockup de la interfaz detalle de personas conocidas o desconocidas

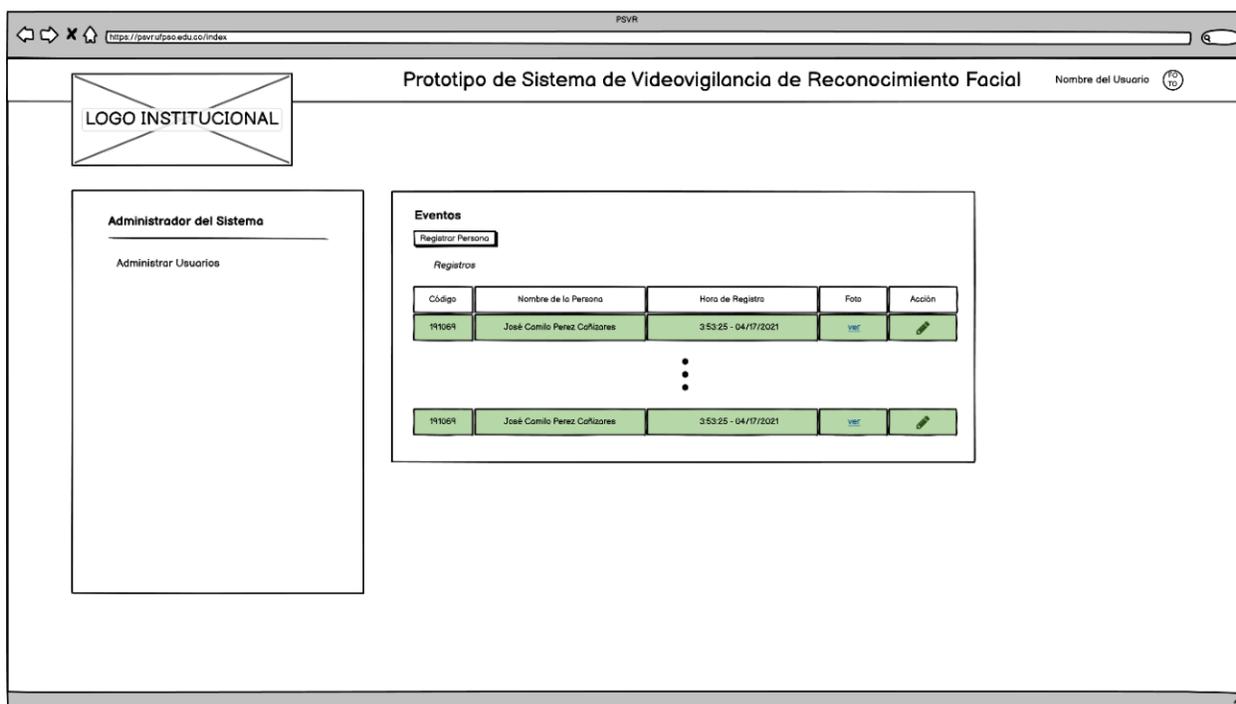


Nota. Realizado por los autores.

Interfaz administrativa para la gestión de usuarios del sistema. Para el registro de las personas que serían identificadas como conocidas dentro del sistema, se planteó una interfaz donde se pudiera realizar el registro de los usuarios por medio de un botón que desplegaría una ventana emergente con los campos apreciados en la **Figura 29**, campos que almacenarían el código asociado a la persona, el nombre de la persona, la hora de almacenamiento del registro, las fotos capturadas para el registro de los rasgos biométricos y la acción de poder editar algún registro si se presentaba algún inconveniente con la información almacenada.

Figura 103

Mockup de la interfaz administrativa para la gestión de usuarios del sistema.

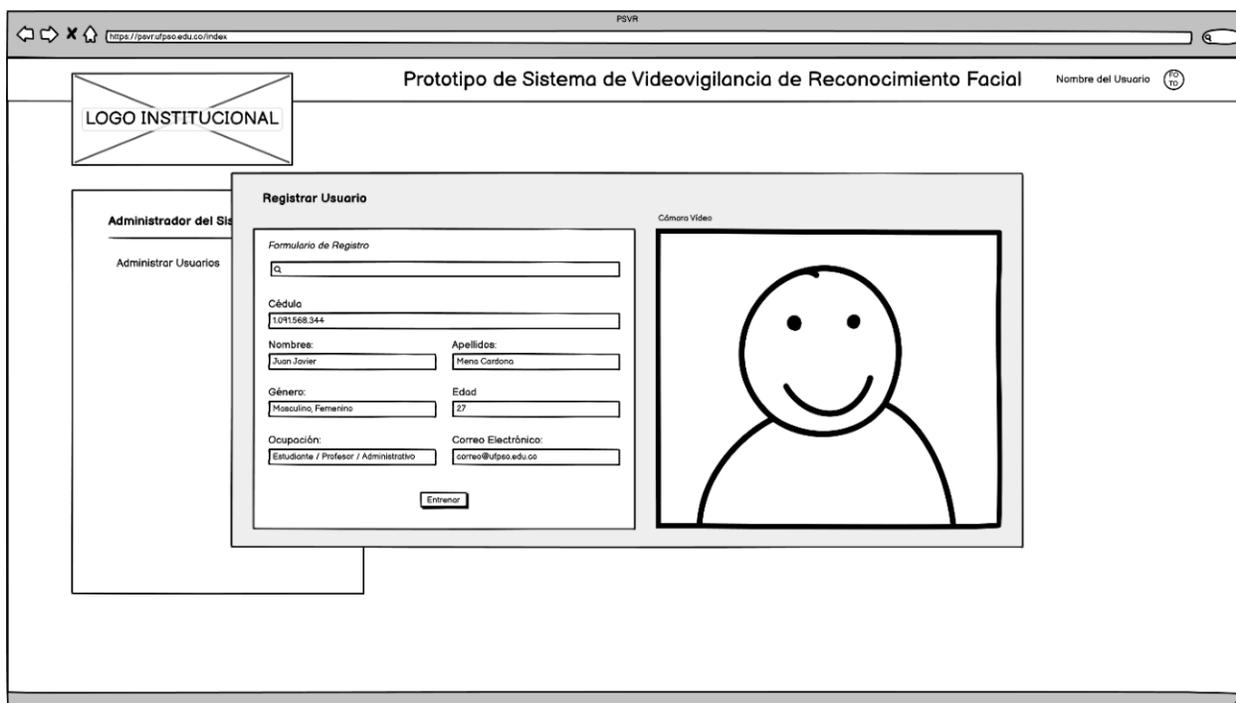


Nota. Realizado por los autores.

Interfaz de registro de usuario. Con la intención de facilitar el proceso de registro, se planteó un diseño con una ventana emergente que presentara un campo de búsqueda para revisar si el usuario ya se encontraba registrado en el sistema, además, sus datos personales como: cédula, nombres, apellidos, género, edad, ocupación y correo electrónico, de igual forma, se propuso la visualización del usuario a través de una cámara de registro para la adquisición de los rasgos biométricos y el botón “Entrenar”, para realizar el proceso de entrenamiento del modelo.

Figura 104

Mockup de la interfaz de registro de usuario



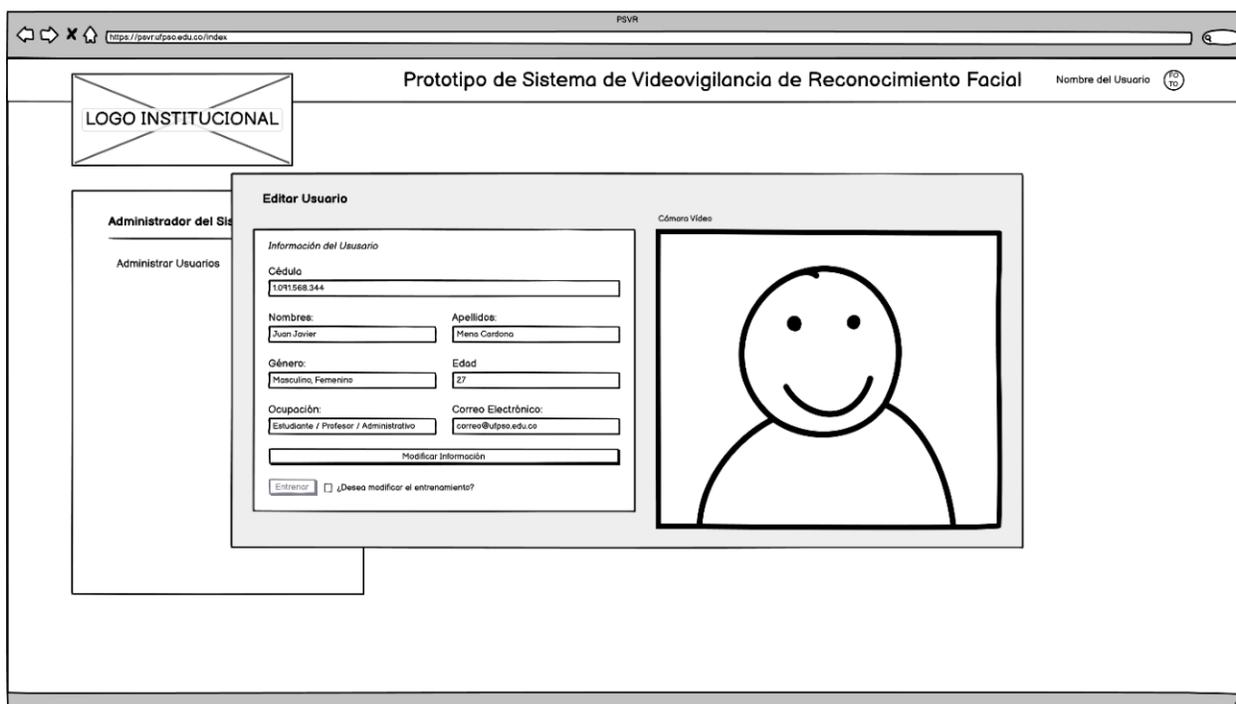
Nota. Realizado por los autores.

Esta ventana era crucial dentro del planteamiento de los diseños, debido a que constituía el proceso más importante del sistema, por lo que debía ser intuitiva para el administrador.

Interfaz de modificación de registros. Como se mencionó anteriormente, se brindó la opción para que el administrador pudiera modificar cualquier tipo de registro si se llegaban a presentar inconsistencias en la información de los usuarios previamente registrados, para ello, se planteó una ventana emergente con las mismas características de la pantalla de “Registro de Usuario”, pero con un botón que permitiera la modificación de la información personal y una casilla de verificación, para que el usuario pudiera habilitar el botón de entrenamiento y así volver a realizar el proceso de captura de los rasgos biométricos.

Figura 105

Mockup de la interfaz de modificación de registros



Nota. Realizado por los autores.

5.2.2.3 Evaluación del prototipo

En la segunda iteración se logró afianzar la construcción del prototipo, se lograron establecer las herramientas definitivas para la construcción del prototipo final y se logró satisfacer el nuevo requerimiento estipulado, se diseñaron las posibles interfaces para la gestión y monitoreo de la información del sistema, y se estructuró el modelo relacional para el registro de los campos planteados en cada interfaz.

5.2.2.3.1 Nivel de satisfacción de los requerimientos. Cumpliendo con la nomenclatura de satisfacción de los requerimientos planteada en la primera iteración, se mantuvieron los mismos niveles de satisfacción de los requerimientos ya planteados, aunque para la segunda iteración, se construyó la base de datos y se planteó un nuevo requerimiento, por lo que su nivel de satisfacción se determinó de la siguiente manera:

Requerimientos no funcionales.

Tabla 40

Nivel de satisfacción del anexo de requerimiento no funcional - Iteración 2

Nivel de Satisfacción de los Requerimientos No Funcionales		
No.	Nivel de Satisfacción	Observación
RA24	S	Se construyó la base de datos en MySQL.
RA28	S	Se establecieron herramientas web para el desarrollo adecuado del sistema.

Nota. Realizado por los autores.

5.2.2.4 Refinamiento del prototipo

Con base en las interfaces propuestas y el modelo relacional planteado, se estimó para la tercera iteración, la construcción de las interfaces en Angular para la gestión del CRUD en Python, comunicándose entre si a través de los WebSockets.

5.2.3 Iteración 3

5.2.3.1 Requerimientos

Para la tercera iteración ya se tenía conceptualizada una arquitectura robusta del sistema en general, aunque se realizaron ajustes a la base de datos con la intención de estructurar todos los procesos del CRUD para la gestión del sistema. Dentro de los requerimientos del sistema no se realizaron modificaciones, por lo que a nivel de hardware como de software se mantuvieron las mismas herramientas.

5.2.3.2 Diseño y construcción

5.2.3.2.1 Hardware. El hardware utilizado para el desarrollo de la tercera iteración fue el mismo que se planteó en las iteraciones anteriores, a diferencia que dentro de la ejecución de esta iteración ya se contaba con la cámara de videovigilancia que había sido expuesta en la primera iteración.

5.2.3.2.2 Software. Dentro de las herramientas de software se utilizaron las herramientas establecidas en la segunda iteración.

5.2.3.2.3 CRUD. Una vez construido el modelo relacional y posteriormente la base de datos en MySQL, se procedió con la estructuración y construcción del sistema para la gestión de la información. Esta etapa del proyecto se desarrolló mediante la creación de métodos y servicios en Angular y Python, buscando establecer una comunicación integral para el almacenamiento de la información, implementado a su vez WebSockets para la transmisión de la información y las imágenes de vídeo, con la intención de realizar la transferencia de imagen y las modificaciones de registros en tiempo real.

Para la construcción del CRUD se crearon funciones específicas dentro de Python que permitieran procesar los datos de forma dinámica, de acuerdo con los parámetros establecidos para cada método. La información se enviaba estructurada en JSON desde el cliente hacia el controlador del servidor UVICORN por medio de servicios en Angular.

Una vez los datos se capturaban, procesaban y ejecutaban según la solicitud de gestión establecida, se retornaban a través de los WebSockets por medio de emisiones.

Figura 106

Funciones de CRUD, validación y notificación

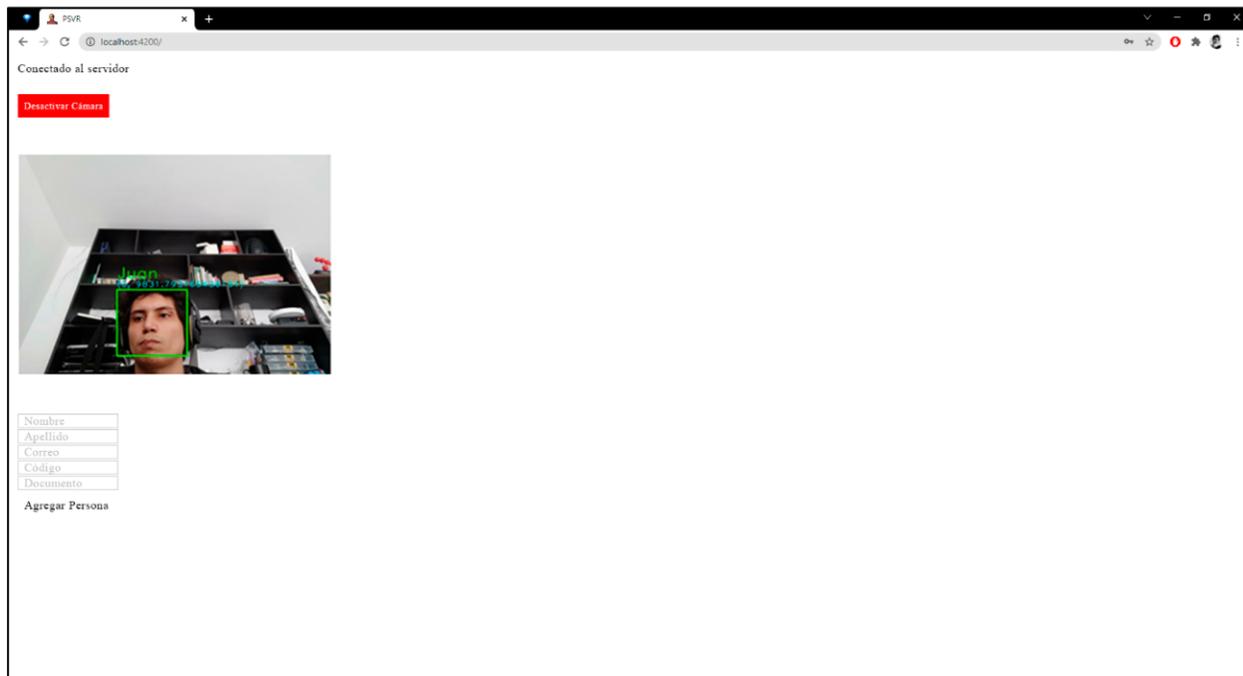
```
> def converter(o): ...  
> def getKeysValuesInsert(parameters): ...  
> def getKeysValuesUpdate(parameters): ...  
> def JSON(parameters, response, sql_type): ...  
> def checkToken(token): ...  
> def tryExcept(parameters, sql, sql_statement): ...  
> def select(parameters, sql_statement): ...  
> def insert(parameters): ...  
> def sendEmail(parameters, email_content, subject): ...  
> def update(parameters): ...  
> def delete(parameters): ...  
> def restore(parameters): ...  
> def query(parameters, condition): ...  
> def unique(table, column, condition, id): ...  
> def token(parameters): ...  
> def recognize(code): ...  
> def recognizeState(code): ...  
> def selectDirectory(code): ...  
> def insertDirectory(code, photo_count): ...  
> def updateDirectory(code, photo_count): ...  
> def socketId(id, socket): ...  
> def logout(id): ...  
> def getSockets(profile): ...  
> def emailHTML(title, message): ...
```

Nota. Realizado por los autores.

Una vez lograda la comunicación por medio de los WebSockets, se realizó una prueba de transmisión de la imagen capturada por la cámara de videovigilancia y la digitación de los campos de entrada, con la finalidad de validar la emisión de la información del cliente hacia el servidor, para su almacenamiento y manipulación.

Figura 107

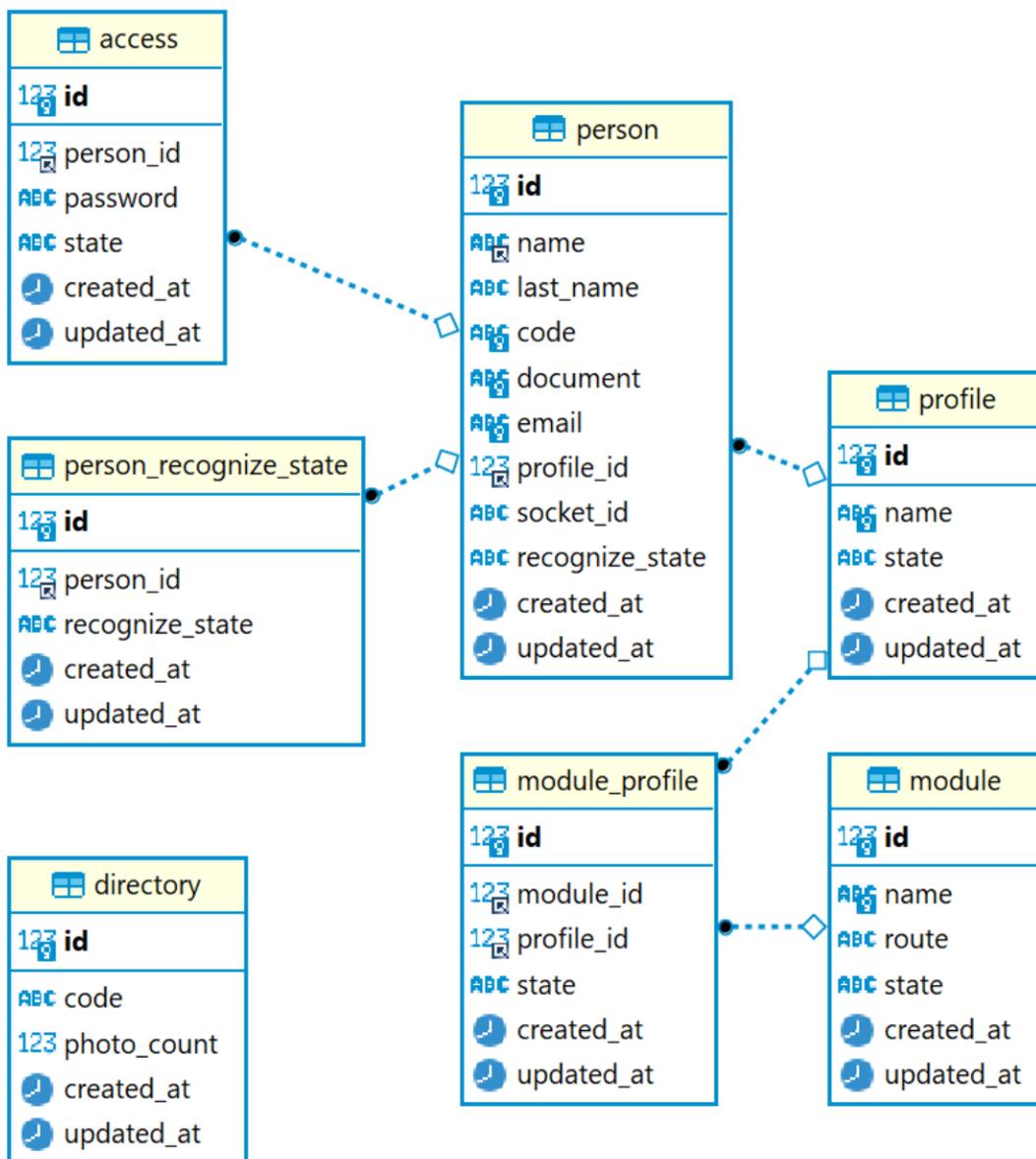
Ejemplo de visualización de la cámara en el navegador con almacenamiento de información en formularios



Nota. Realizado por los autores.

5.2.3.2.4 Modelo relacional del sistema - Modificado.**Figura 108**

Modelo relacional del sistema en la iteración 3.



Nota. Ajustes al modelo relacional planteado. Realizado por los autores.

5.2.3.2.5 Detalle de modificación de las tablas del modelo relacional.

Tabla 41*Detalle de las tablas del modelo relacional - Iteración 3*

Detalle de las Tablas del Modelo Relacional		
Nombre de la Tabla	Descripción (Modificación)	Relación
“directory”	Permite gestionar la información del directorio temporal, almacenando un identificador, la cantidad de fotos capturadas y la fecha de creación del registro.	No posee relación.

Nota. Realizado por los autores.

Las modificaciones realizadas al modelo relacional se basaron en la idea de llevar un proceso de control de la cantidad de fotos almacenadas de determinada persona dentro de la base de datos.

Con las conexiones entre cliente y servidor establecidas y el almacenamiento de la información en la base de datos estipulada, se procedió con la construcción de las interfaces planteadas en los mockups.

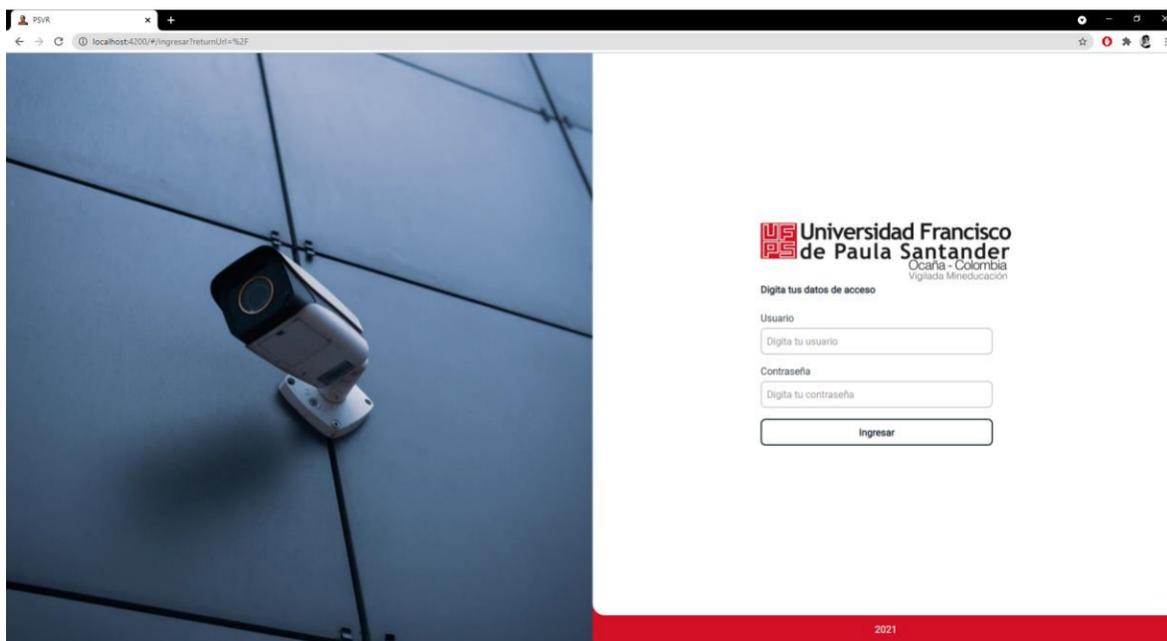
Siguiendo el patrón de diseño establecido en la universidad, se manejaron los mismos logotipos y la misma paleta de colores, siendo congruentes con los sistemas actualmente implementados en la universidad, guiados bajo los mismos criterios de diseño.

5.2.3.2.6 Interfaces de Usuario.

Interfaz de inicio de sesión. La interfaz de inicio de sesión se construyó en primera instancia siguiendo la misma estructura planteada en los mockups, aunque se realizaron pequeñas modificaciones, donde no se podía reestablecer la contraseña como se había planteado, sino que el administrador era la única persona que tenía la responsabilidad de ejecutar dichas acciones.

Figura 109

Interfaz de inicio de sesión

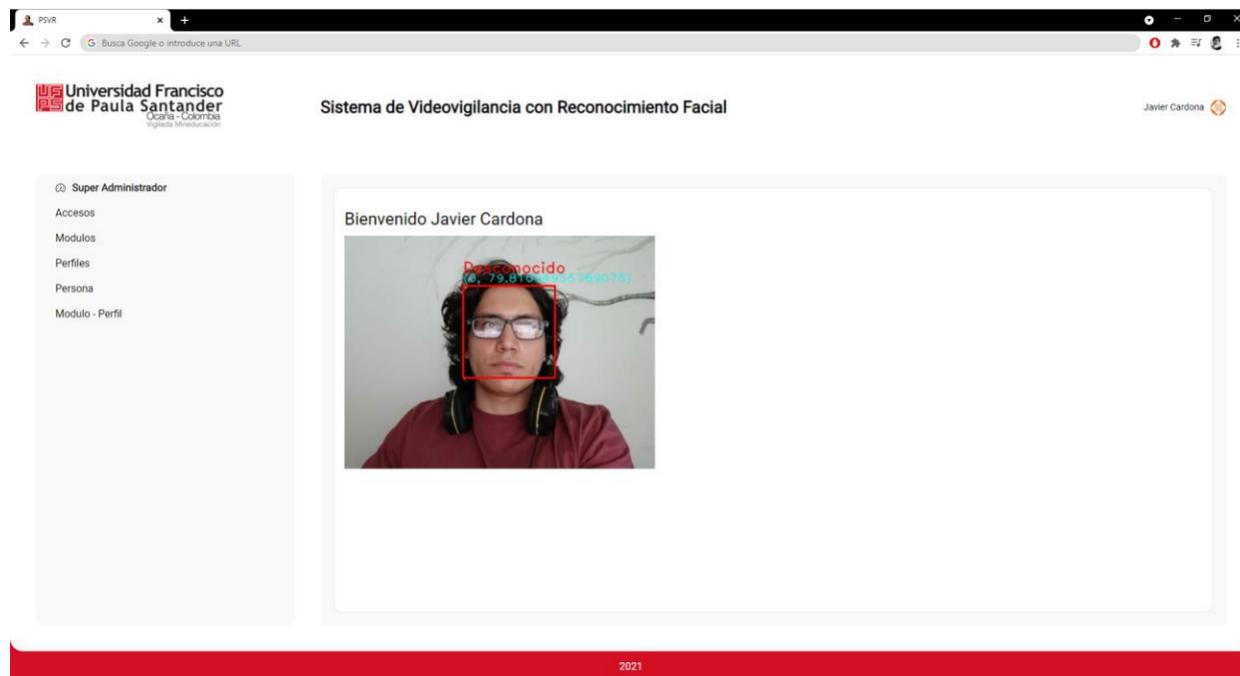


Nota. Captura extraída del sistema. Realizado por los autores.

Interfaz de monitoreo. Una vez el usuario accedía al sistema por medio de las credenciales, podía visualizar la “Interfaz de Monitoreo” donde se exponían las imágenes de todas las cámaras de monitoreo asociadas al prototipo, en este caso en particular solo se visualizaba una.

Figura 110

Interfaz de monitoreo



Nota. Realizado por los autores.

A su vez, el usuario tenía la posibilidad de interactuar con el sistema por medio del menú lateral donde se presentaban las siguientes opciones:

- **Nombre del Perfil:** Perfil asociado al usuario con acceso al sistema; en primera instancia se definieron los perfiles de Super Administrador, Inspector de Cámaras, Administrador Técnico, Administrador del Sistema, Docente y Estudiante.
- **Accesos:** Módulo que permite la gestión de la información de los accesos al sistema.
- **Módulos:** Módulo que permite la gestión de la información de los módulos del sistema.
- **Perfiles:** Módulo que permite la gestión de los diferentes roles del sistema.
- **Persona:** Módulo que permite la gestión de la información de las personas registradas en el sistema.
- **Modulo – Perfil:** Módulo que permite la gestión y asociación de la información del módulo respecto al perfil.

Las opciones del menú podían estar expuestas a cambios por parte del Super Administrador, pero por defecto el sistema contaba con las opciones mencionadas anteriormente.

Interfaz de accesos. Al clicar en el módulo de “Accesos” el usuario podía visualizar el botón de “Crear” y la información de los registros de los usuarios con acceso al sistema, tales como:

Figura 111

Interfaz de accesos

Universidad Francisco de Paula Santander
Vigilancia MinEducación

Sistema de Videovigilancia con Reconocimiento Facial

Javier Cardona

Super Administrador

Accesos

Modulos

Perfiles

Persona

Modulo - Perfil

CREAR

Nombre	Apellido	Código	Documento	Correo	Estado	Acción
Javier	Cardona	191061	1022370118	axroxex@gmail.com	HABILITADO	🗑️
Juan	Mena	191047	12345678	jupamesapemo@gmail.com	HABILITADO	🗑️

2021

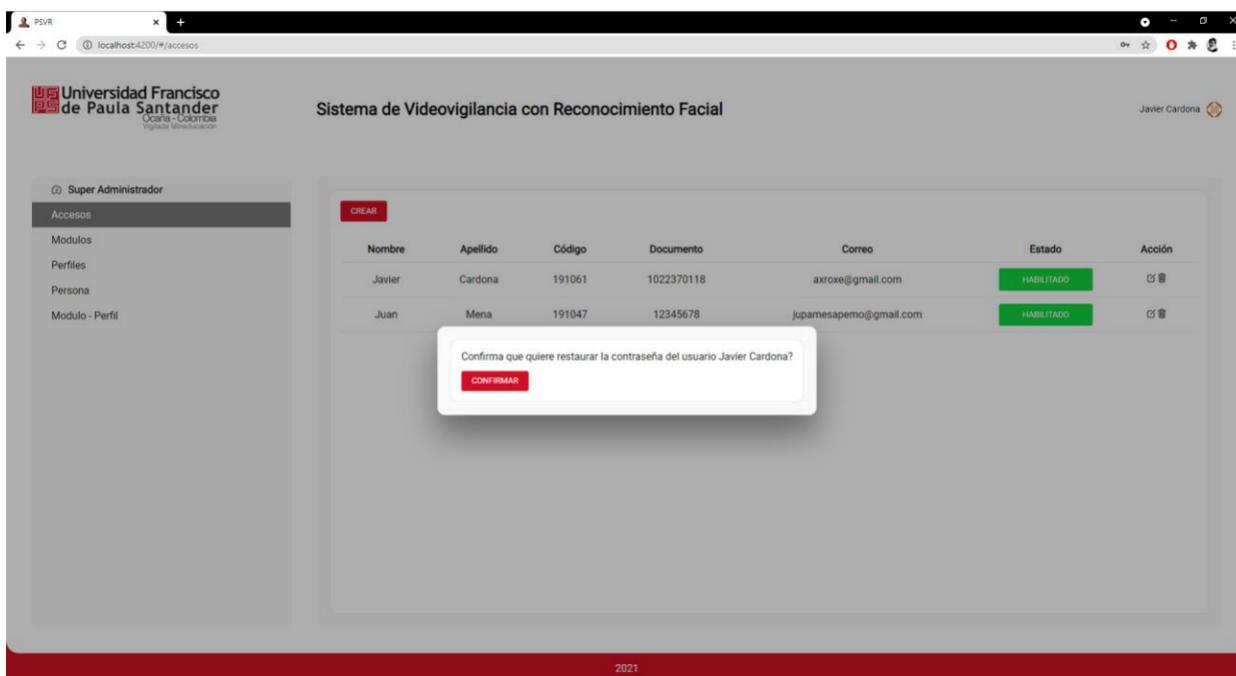
Nota. Realizado por los autores.

- **Nombre:** Nombres del usuario.
- **Apellido:** Apellidos del usuario.
- **Código:** Código del usuario registrado en la institución.
- **Documento:** Documento de identificación del usuario.
- **Correo:** Correo electrónico registrado en la institución.
- **Estado:** Estados de acceso al sistema. Dentro de los estados posibles se tenían las opciones de “Habilitado” (resaltado en color verde) y “Deshabilitado” (resaltado en color rojo). De acuerdo con el estado del registro, se otorgaba o denegaba el acceso al sistema.
- **Acción:** Acciones de modificación del registro de accesos. Dentro de las opciones se encontraban:

- **Restaurar Contraseña** (🔑): Opción que desplegaba una ventana emergente con un mensaje de confirmación y un botón de “Confirmar”, con la finalidad de reestablecer la contraseña por medio del correo electrónico asociado a la cuenta del usuario con acceso al sistema.

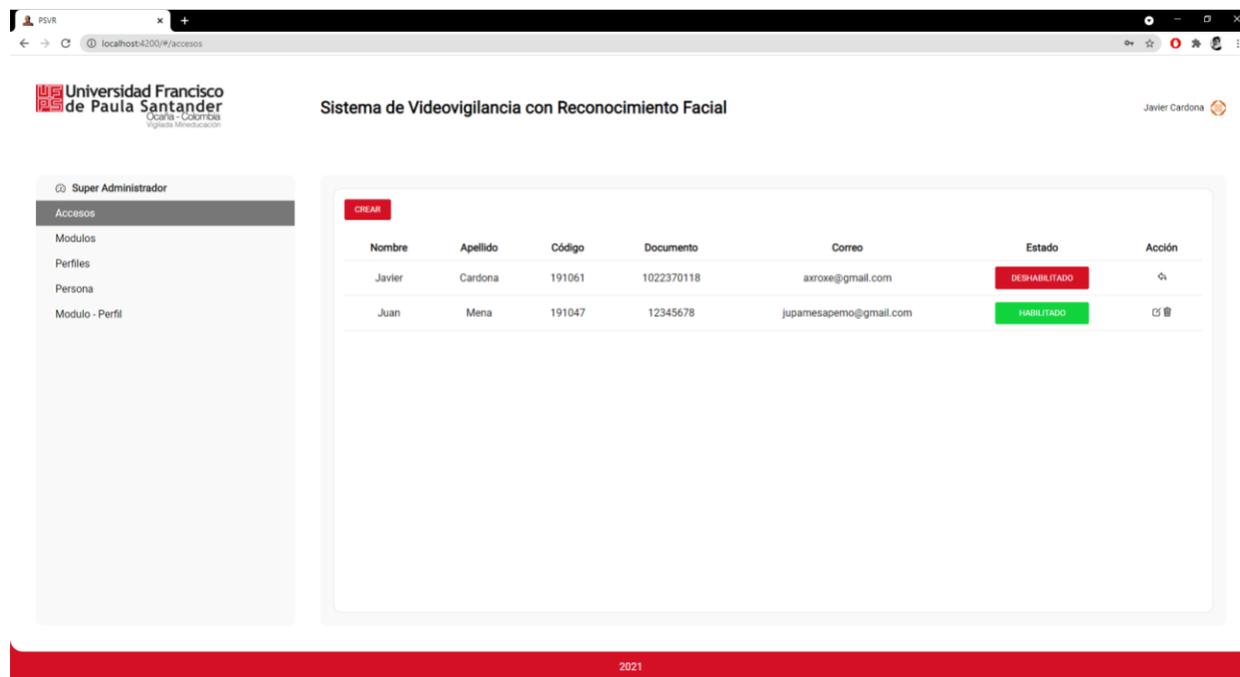
Figura 112

Interfaz de validación para el cambio de contraseña del usuario



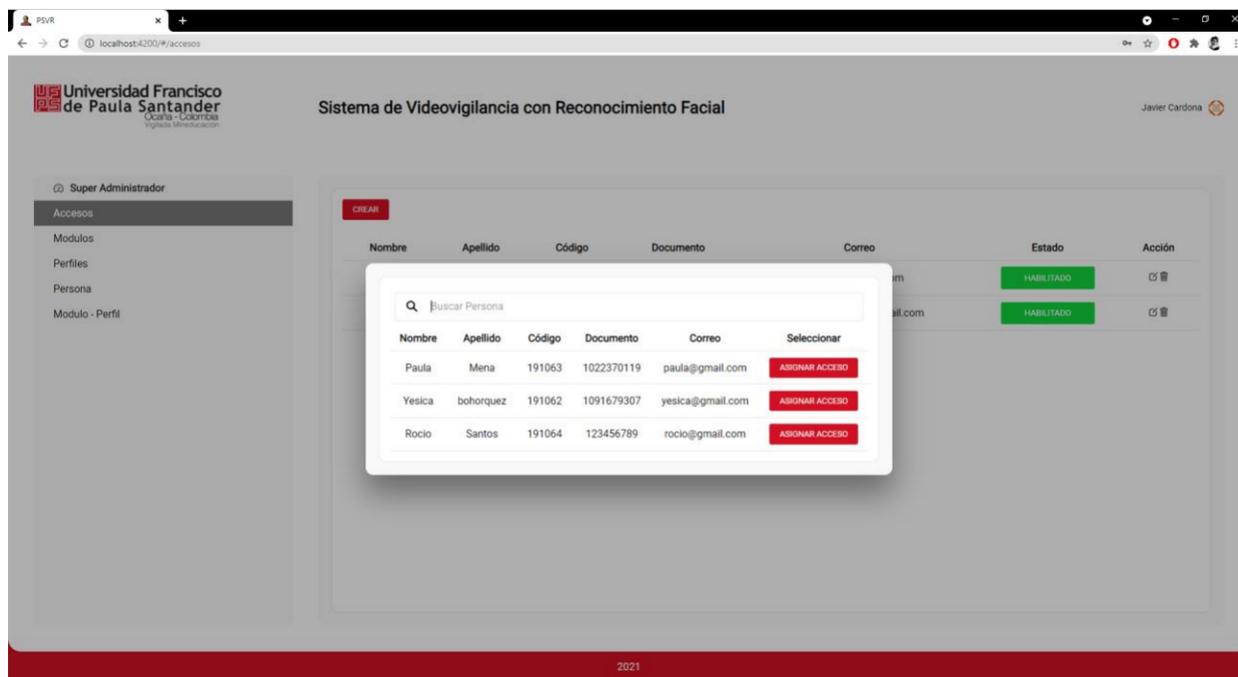
Nota. Realizado por los autores.

- **Inhabilitar Acceso** (🗑️): Opción que deshabilitaba el acceso del usuario registrado en el sistema.

Figura 113*Interfaz de acceso deshabilitado*

Nota. Realizado por los autores.

- **Habilitar Acceso** (🔑): Opción que habilitaba el acceso del usuario registrado en el sistema.
- **Botón Crear:** Botón que permitía desplegar una ventana emergente con un buscador, el registro de los usuarios en el sistema y el botón de “Asignar Acceso”. El buscador tenía la finalidad de realizar búsquedas específicas dentro de los múltiples registros de las personas para garantizar el acceso a determinado usuario, de igual forma, el botón de “Asignar Acceso”, confirmaba el acceso, asignando el estado “Habilitado”.

Figura 114*Interfaz de búsqueda y asignación de accesos*

Nota. Realizado por los autores.

Interfaz de módulos. De igual forma que la opción de “Accesos”, el usuario al clicar sobre la opción de “Módulos”, tenía la posibilidad de gestionar los módulos del sistema, modificando el nombre, ruta y estados de los mismos por medio de los botones de “Acción”.

Figura 115*Interfaz de módulos*

Universidad Francisco de Paula Santander
Vicerrectoría de Investigación y Desarrollo Tecnológico

Sistema de Videovigilancia con Reconocimiento Facial

Javier Cardona

Super Administrador

Accesos

Modulos

Perfiles

Persona

Modulo - Perfil

CREAR

#	Nombre	Ruta	Estado	Acción
1	Persona	/persona	HABILITADO	✎ 🗑
2	Modulos	/modulo	HABILITADO	✎ 🗑
3	Perfiles	/perfil	HABILITADO	✎ 🗑
4	Accesos	/accesos	HABILITADO	✎ 🗑

Anterior 1 2 Sigiente

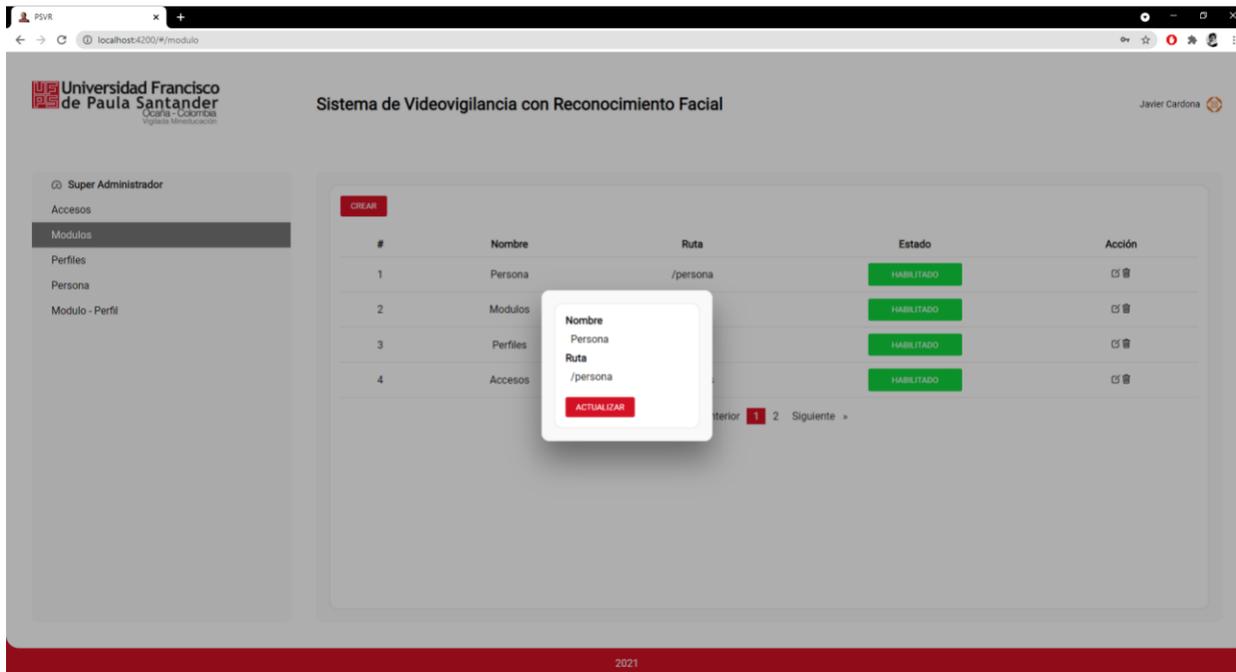
2021

Nota. Realizado por los autores.

- **Nombre:** Nombre del módulo.
- **Ruta:** Nombre de la ruta de redireccionamiento del módulo.
- **Estado:** Estados del módulo del sistema. Dentro de los estados posibles se tenían las opciones de “Habilitado” (resaltado en color verde) y “Deshabilitado” (resaltado en color rojo). De acuerdo con el estado del registro, se visualizaba u ocultaba el módulo en el menú del sistema.
- **Acción:** Acciones de modificación del registro de los módulos. Dentro de las opciones se encontraban:
 - **Editar Nombre** (✎): Opción que desplegaba una ventana emergente con campos autocompletados con la información del “Nombre” y “Ruta” del módulo. A su vez, se visualizaba el botón de “Actualizar”, para realizar el proceso de actualización de la información del registro.

Figura 116

Interfaz de modificación de los módulos

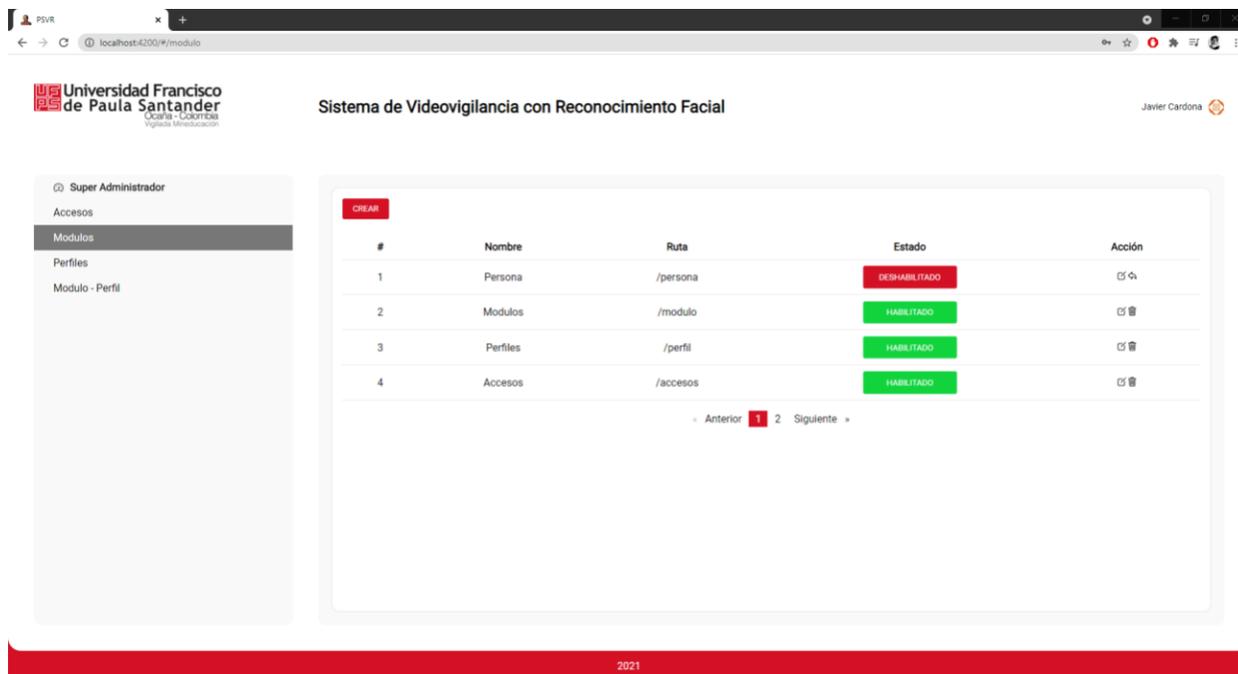


Nota. Realizado por los autores.

- **Inhabilitar Módulo** (🗑️): Opción que deshabilitaba el módulo para ser visualizado dentro del sistema.

Figura 117

Interfaz de módulos deshabilitados

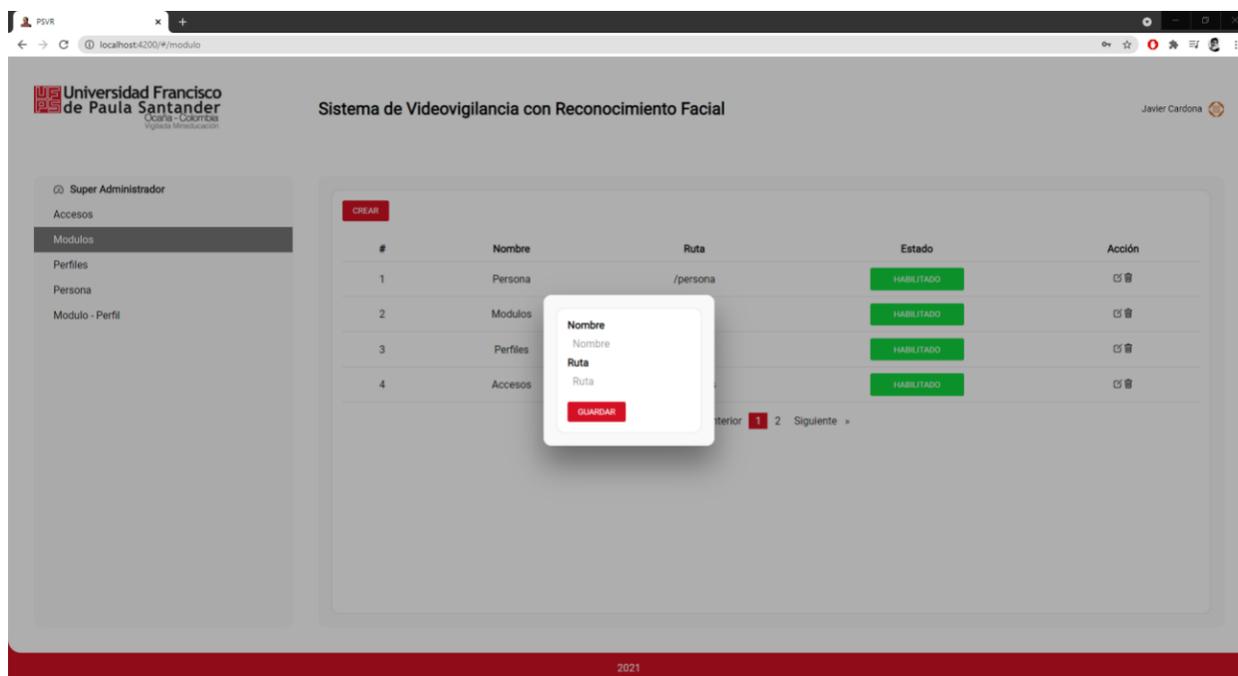


Nota. Realizado por los autores.

- **Habilitar Módulo** (🔗): Opción que habilitaba el módulo para ser visualizado dentro del sistema.
- **Botón Crear:** Botón que permitía desplegar una ventana emergente con los campos de “Nombre”, “Ruta” y el botón de “Guardar” para realizar el registro del nuevo módulo.

Figura 118

Interfaz de creación de módulos



Nota. Realizado por los autores.

Interfaz de perfiles. En la opción de “Perfiles” el administrador podía gestionar la información de los perfiles creados dentro del sistema, modificando el nombre y los estados de los mismos por medio de los botones de “Acción”.

Figura 119

Interfaz de perfiles

Universidad Francisco de Paula Santander
Vigilancia MinEducación

Sistema de Videovigilancia con Reconocimiento Facial

Javier Cardona

Super Administrador

Accesos

Modulos

Perfiles

Persona

Modulo - Perfil

CREAR

#	Nombre	Estado	Acción
1	Super Administrador	HABILITADO	🗑️
2	Inspector de Cámaras	HABILITADO	🗑️
3	Administrador Técnico	HABILITADO	🗑️
4	Administrador del Sistema	HABILITADO	🗑️

Anterior 1 2 Sigiente

2021

Nota. Realizado por los autores.

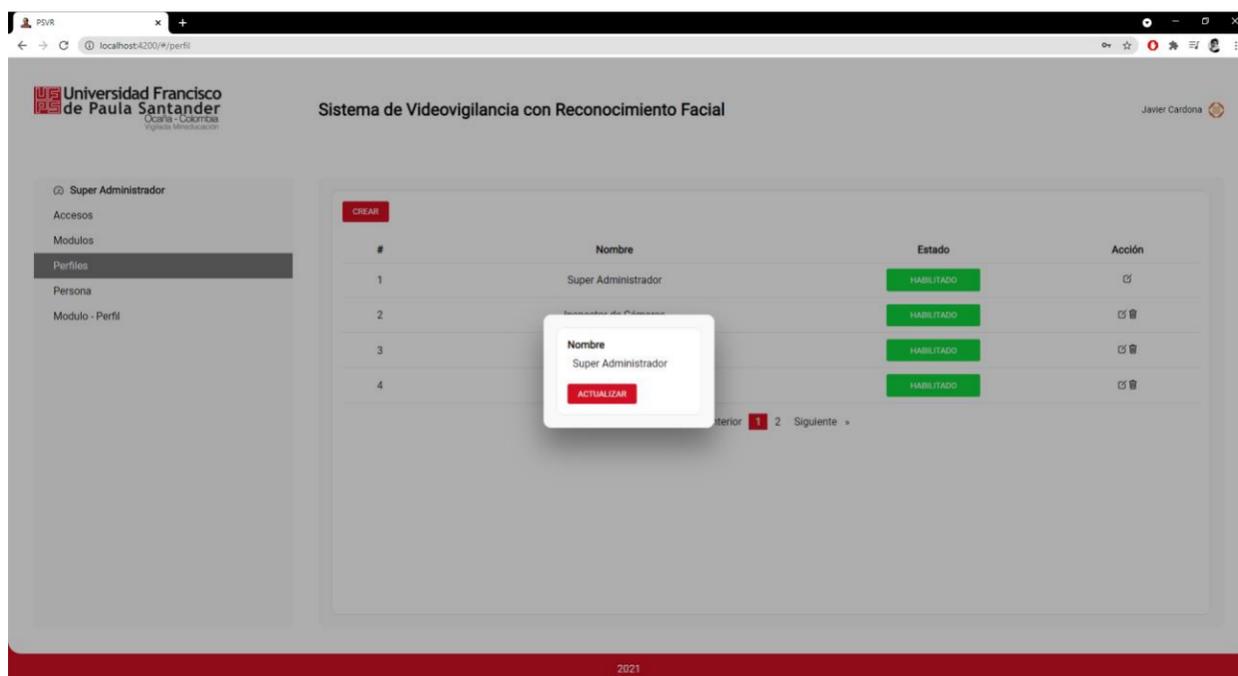
Dentro de las posibilidades de acción, se realizó una restricción al perfil de Super Administrador, con la finalidad de reducir los privilegios de inhabilitar su propio perfil, por motivos de integridad, seguridad y coherencia del registro.

- **Nombre:** Nombre del perfil.
- **Estado:** Estados del perfil del sistema. Dentro de los estados posibles se tenían las opciones de “Habilitado” (resaltado en color verde) y “Deshabilitado” (resaltado en color rojo). De acuerdo con el estado del registro, se habilitaba o inhabilitaba el perfil dentro del sistema.
- **Acción:** Acciones de modificación del registro de los perfiles. Dentro de las opciones se encontraban:

- **Editar Nombre** (✎): Opción que desplegaba una ventana emergente con campos autocompletados con la información del “Nombre” del perfil. A su vez, se visualizaba el botón de “Actualizar”, para realizar el proceso de actualización de la información del registro.

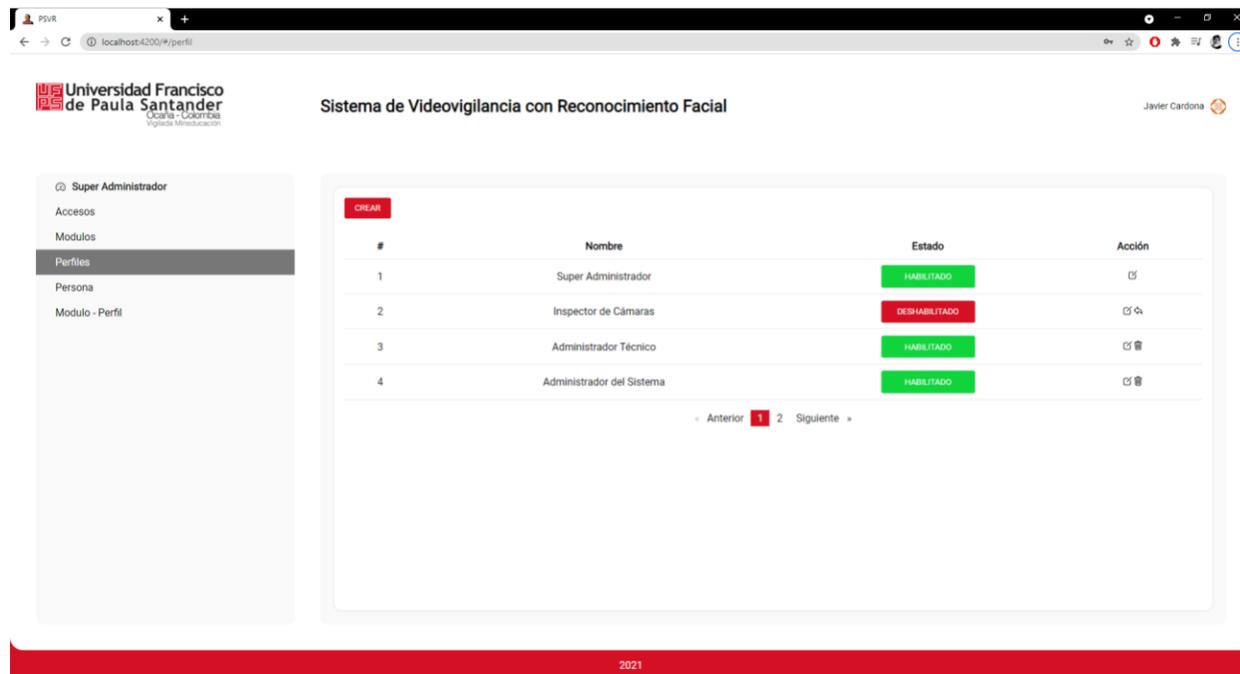
Figura 120

Interfaz de modificación de perfiles



Nota. Realizado por los autores.

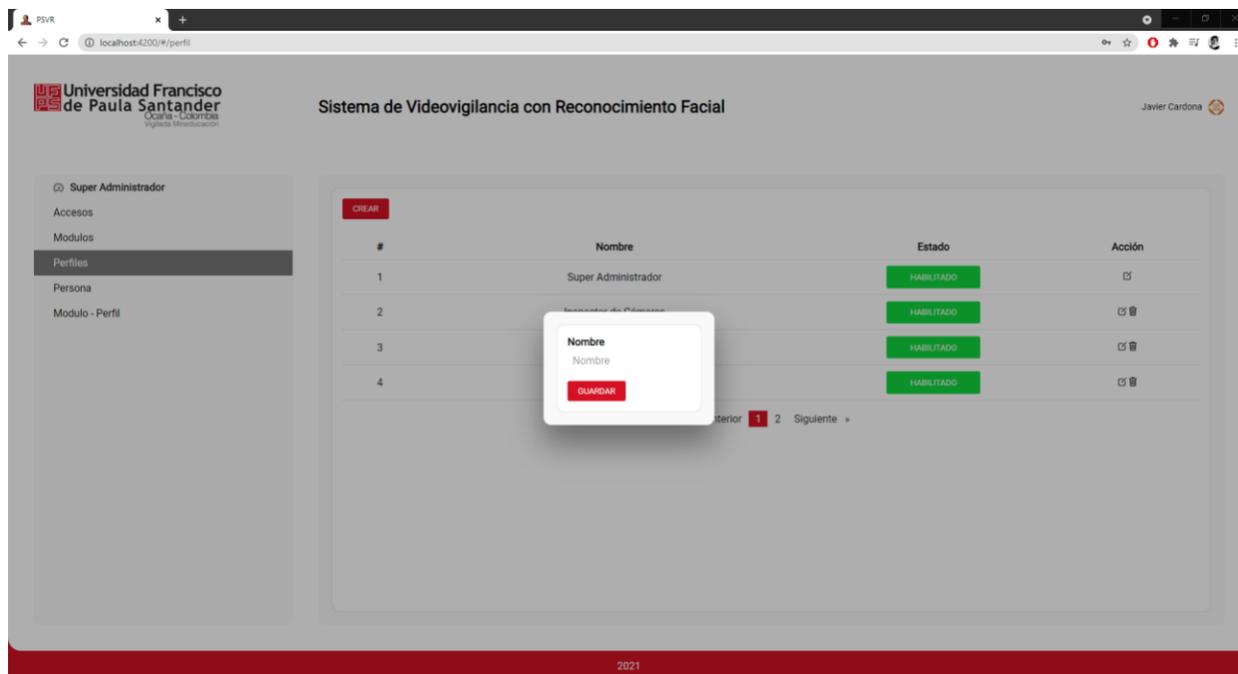
- **Inhabilitar Perfil** (🗑️): Opción que deshabilitaba el perfil para ser visualizado dentro del sistema.

Figura 121*Interfaz de perfiles deshabilitados*

Nota. Realizado por los autores.

- **Habilitar Perfil** (🔄): Opción que habilitaba el perfil para ser visualizado dentro del sistema.
- **Botón Crear:** Botón que permitía desplegar una ventana emergente con los campos de “Nombre” y el botón de “Guardar” para realizar el registro del nuevo perfil.

Figura 122*Interfaz de creación de perfiles*



Nota. Realizado por los autores.

Interfaz de personas. Igual como en las opciones descritas anteriores, el administrador al clicar en la opción “Persona”, podía gestionar toda la información relacionada con las personas registradas dentro del sistema, por medio del:

Figura 123

Interfaz de personas

Universidad Francisco de Paula Santander
Vigilancia MinEducación

Sistema de Videovigilancia con Reconocimiento Facial

Javier Cardona

Super Administrador

Accesos

Modulos

Perfiles

Persona

Modulo - Perfil

CREAR

Nombre	Apellido	Código	Documento	Correo	Perfil	Acción
Javier	Cardona	191061	1022370118	axroxe@gmail.com	Super Administrador	✎ 🗑
Juan	Mena	191047	12345678	jupamesapemo@gmail.com	Super Administrador	✎ 🗑
Paula	Mena	191063	1022370119	paula@gmail.com	Inspector de Cámaras	✎ 🗑
Yesica	bohorquez	191062	1091679307	yesica@gmail.com	Docente	✎ 🗑
Rocio	Santos	191064	123456789	rocio@gmail.com	Administrador del Sistema	✎

2021

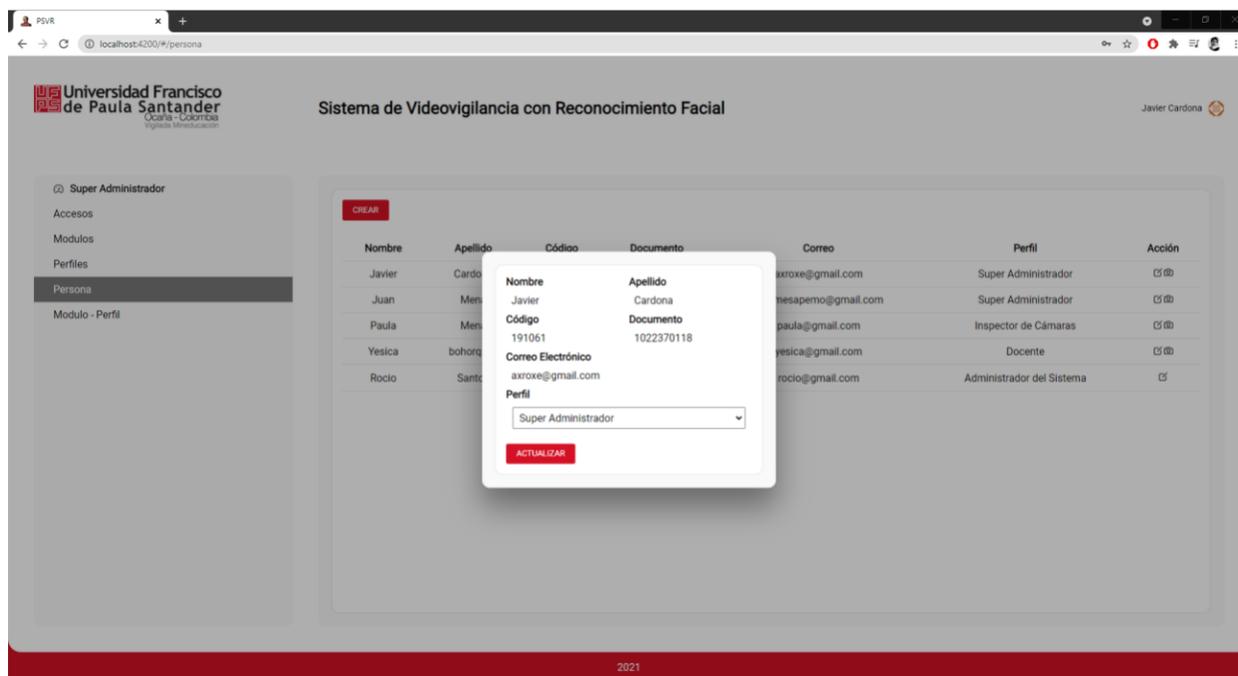
Nota. Realizado por los autores.

- **Nombre:** Nombres del usuario.
- **Apellido:** Apellidos del usuario.
- **Código:** Código del usuario registrado en la institución.
- **Documento:** Documento de identificación del usuario.
- **Correo:** Correo electrónico registrado en la institución.
- **Perfil:** Perfil asociado al usuario.
- **Acción:** Acciones de modificación del registro de los usuarios. Dentro de las opciones se encontraban:
 - **Editar Persona** (✎): Opción que desplegaba una ventana emergente con campos autocompletados con la información del “Nombre”, “Apellido”, “Código”, “Documento” y “Correo Electrónico” de la persona, a su vez, un listado de opciones con los “Perfiles” registrados y un botón de

“Actualizar”, para realizar el proceso de actualización de la información del registro.

Figura 124

Interfaz de modificación de la persona

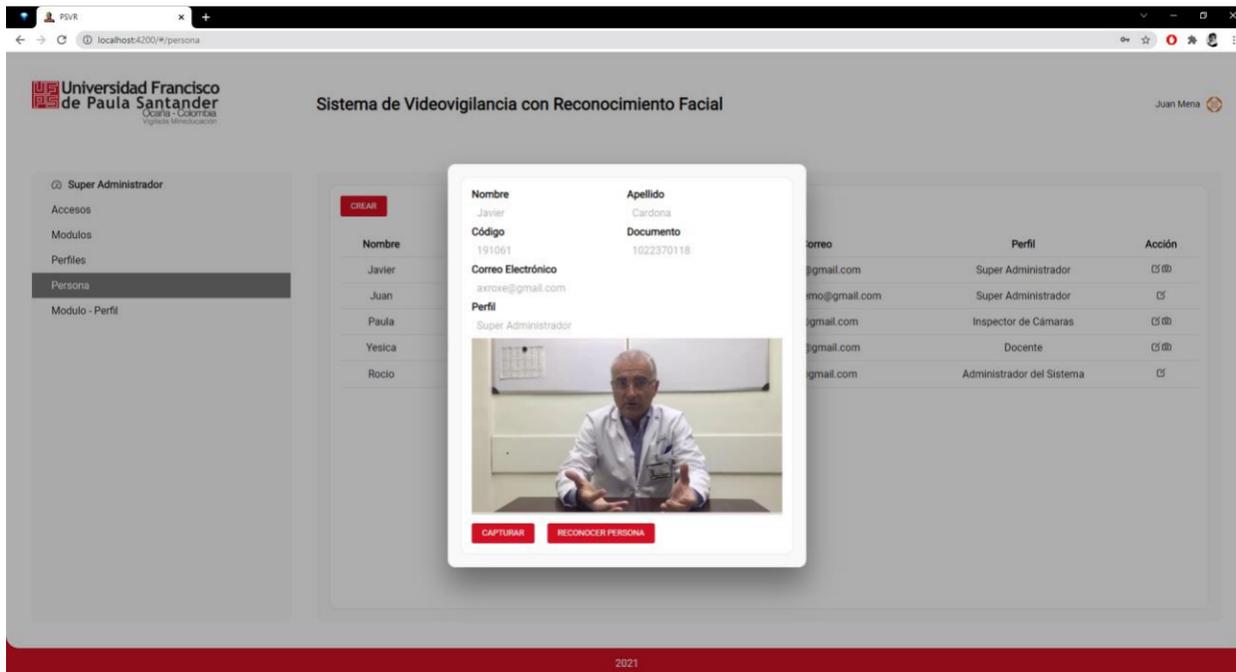


Nota. Realizado por los autores.

- **Capturar Rasgos Biométricos** (📷): Opción que desplegaba una ventana emergente con campos autocompletados con el “Nombre”, “Apellido”, “Código”, “Documento”, “Correo Electrónico” y “Perfil” de la persona, también, un contenedor con las imágenes capturadas por la cámara de registro de las características faciales y los botones “Capturar” y “Reconocer Persona”.

Figura 125

Interfaz de despliegue de la cámara de registro

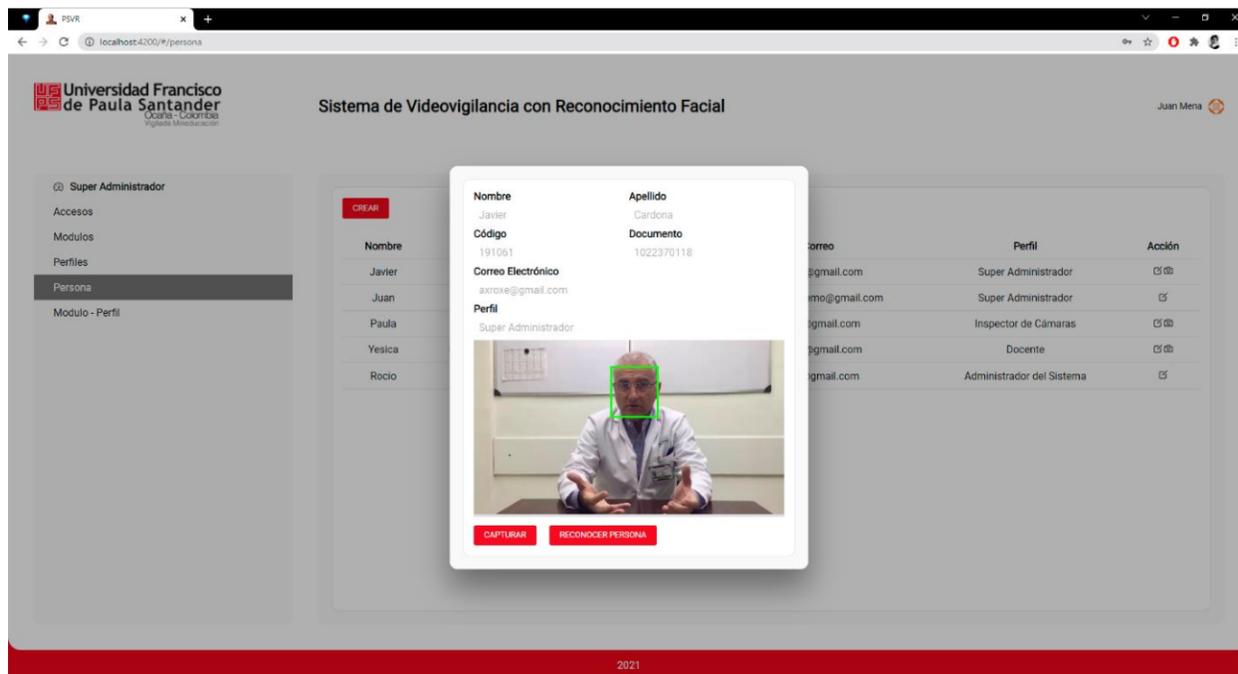


Nota. Imagen ilustrativa de la ventana emergente con las opciones de registro de los rasgos biométricos. Realizado por los autores.

La idea principal de la ventana emergente era permitirle al administrador visualizar la información previamente registrada de la persona, la proyección de vídeo de la cámara de registro y el accionar para el proceso de extracción de rasgos biométricos a través de las funciones de los botones. Una vez la persona a registrar se encontraba ubicada en frente de la cámara de registro, el administrador debía clicar en el botón “Capturar” para ejecutar las funciones de detección y extracción del rostro, sobreponiéndose un recuadro verde sobre la imagen de vídeo, resaltando la ubicación del rostro de donde se extraían las fotos de muestra.

Figura 126

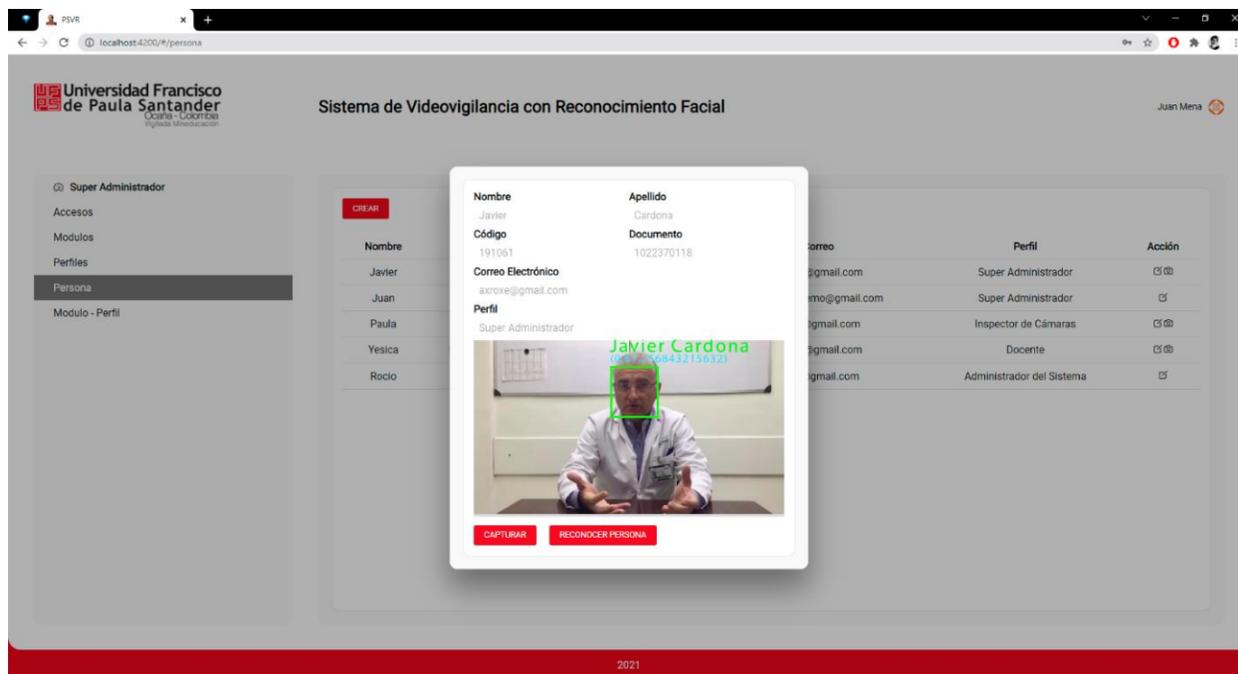
Interfaz de detección y extracción del rostro



Nota. Realizado por los autores.

Figura 127

Asociación de etiquetas en el arreglo matricial con Numpy y entrenamiento del modelo

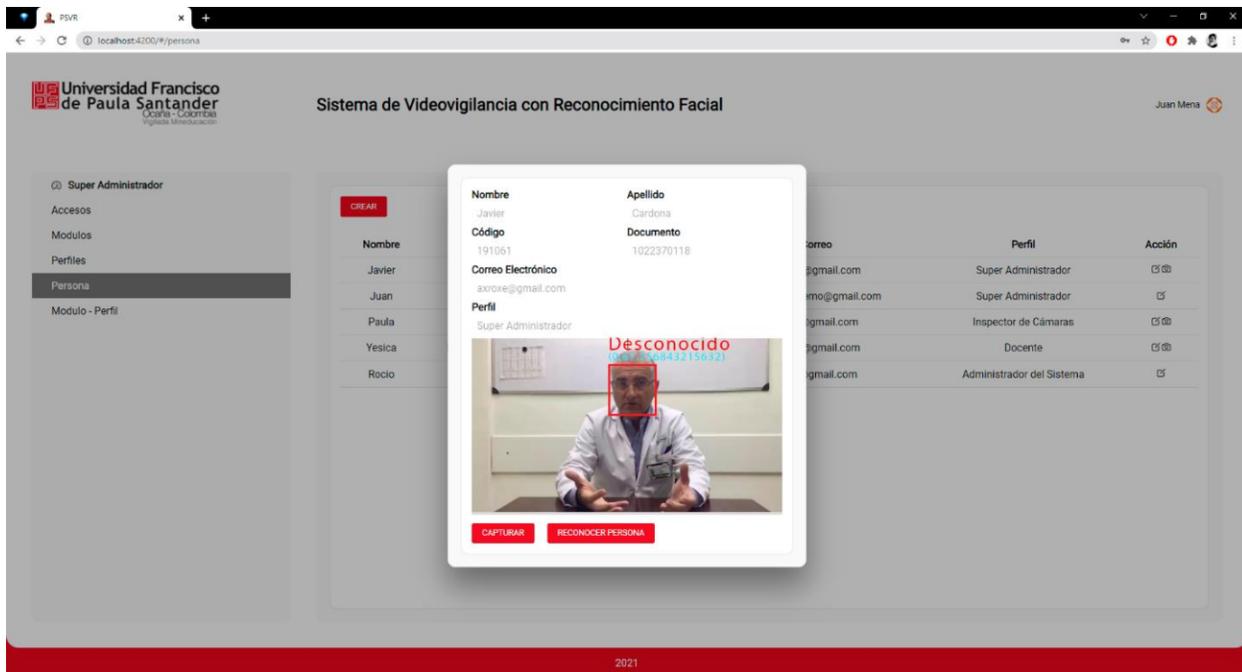


Nota. Los datos referenciados en la figura son datos de prueba del funcionamiento. Realizado por los autores.

Si el umbral de detección establecido no se sobrepasaba, el rostro sería desconocido para el sistema, por tal motivo, el administrador debía realizar de nuevo una captura de los rasgos biométricos de la persona, alimentando nuevamente los ficheros con las imágenes previamente recolectadas del rostro de la persona. El no sobrepasar el umbral estaba determinado por agentes externos que afectaban la calidad de las imágenes tales como: luminosidad del entorno, distorsión de la imagen, ruido, entre otras alteraciones que impedían el reconocimiento en la primera captura de imágenes.

Figura 129

Interfaz de desaprobación del entramiento de reconocimiento facial

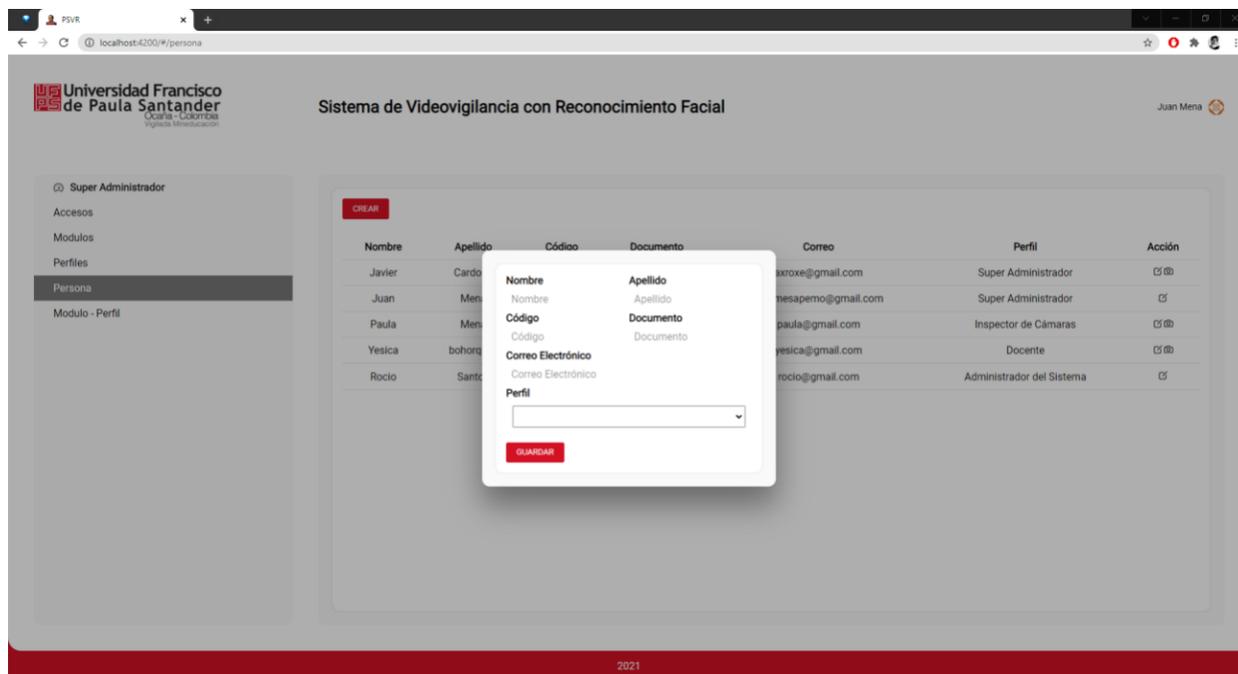


Nota. Realizado por los autores.

- **Botón Crear:** Botón que permitía desplegar una ventana emergente con los campos de “Nombre”, “Apellido”, “Código”, “Documento” y “Correo Electrónico”, a su vez, un listado de opciones con los “Perfiles” registrados y el botón de “Guardar” para realizar el registro del nuevo perfil.

Figura 130

Interfaz de creación de personas



Nota. Realizado por los autores.

Interfaz módulo – perfil. Por último, al clicar sobre la opción de “Modulo – Perfil” el administrador podía visualizar la información de los nombres de los módulos y los perfiles registrados previamente en el sistema, a su vez el estado del registro, pudiendo inhabilitarlos por medio de los botones de “Acción”.

Figura 131

Interfaz módulo - perfil

#	Nombre	Perfil	Estado	Acción
1	Accesos	Super Administrador	HABILITADO	🗑️
2	Modulos	Super Administrador	HABILITADO	🗑️
3	Perfiles	Super Administrador	HABILITADO	🗑️
4	Persona	Super Administrador	HABILITADO	🗑️
5	Persona	Administrador del Sistema	HABILITADO	🗑️
6	Modulos	Administrador del Sistema	HABILITADO	🗑️
7	Perfiles	Administrador Técnico	HABILITADO	🗑️
8	Modulo - Perfil	Super Administrador	HABILITADO	🗑️

Nota. Realizado por los autores.

- **Nombre:** Nombre del módulo.
- **Perfil:** Nombre del perfil asociado al módulo.
- **Estado:** Estados del vínculo entre el módulo y el perfil dentro del sistema. Dentro de los estados posibles se tenían las opciones de “Habilitado” (resaltado en color verde) y “Deshabilitado” (resaltado en color rojo). De acuerdo con el estado del registro, se habilitaba o inhabilitaba la asociación entre el módulo y el perfil dentro del sistema.
- **Acción:** Acciones de inhabilitación del registro del módulo con el perfil. Dentro de las opciones se encontraban:
 - **Inhabilitar Módulo - Perfil** (🗑️): Opción que deshabilitaba el módulo asociado al perfil para ser visualizado dentro del sistema por determinado perfil.

Figura 132

Interfaz de módulos deshabilitados asociados con el perfil

#	Nombre	Perfil	Estado	Acción
1	Accesos	Super Administrador	DESHABILITADO	🔍
2	Modulos	Super Administrador	HABILITADO	🗑️
3	Perfiles	Super Administrador	HABILITADO	🗑️
4	Persona	Super Administrador	HABILITADO	🗑️
5	Persona	Administrador del Sistema	HABILITADO	🗑️
6	Modulos	Administrador del Sistema	HABILITADO	🗑️
7	Perfiles	Administrador Técnico	HABILITADO	🗑️
8	Modulo - Perfil	Super Administrador	HABILITADO	🗑️

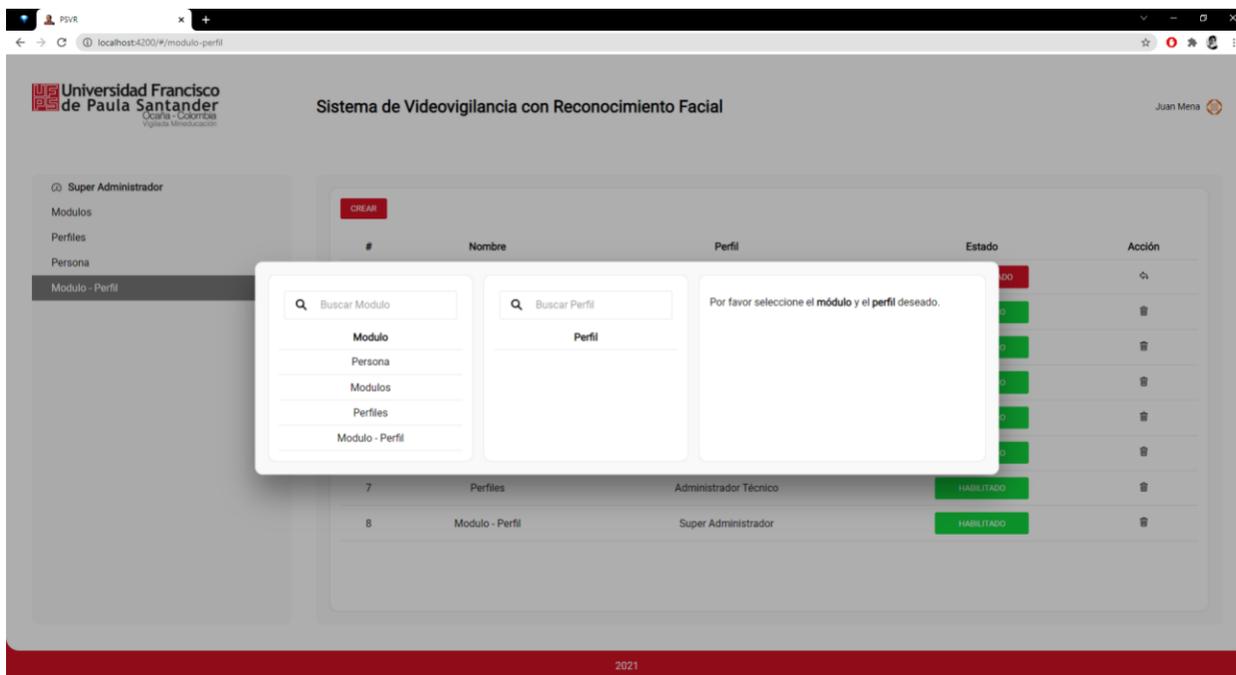
Nota. Realizado por los autores.

- **Habilitar Perfil** (🔍): Opción que habilitaba el módulo asociado al perfil para ser visualizado dentro del sistema por determinado perfil.
- **Botón Crear:** Botón que permitía desplegar una ventana emergente con los siguientes segmentos:
 - **Segmento de Módulos:** Segmento con un buscador de módulos y un listado de los módulos registrados en el sistema; el administrador podía realizar la búsqueda del módulo o clicar directamente sobre el listado de módulos, para seleccionar la opción de módulo que deseaba asociar con el

perfil, una vez realizada la acción, el módulo seleccionado se resaltaba en color rojo.

Figura 133

Interfaz de búsqueda de módulos y perfiles para su asociación.

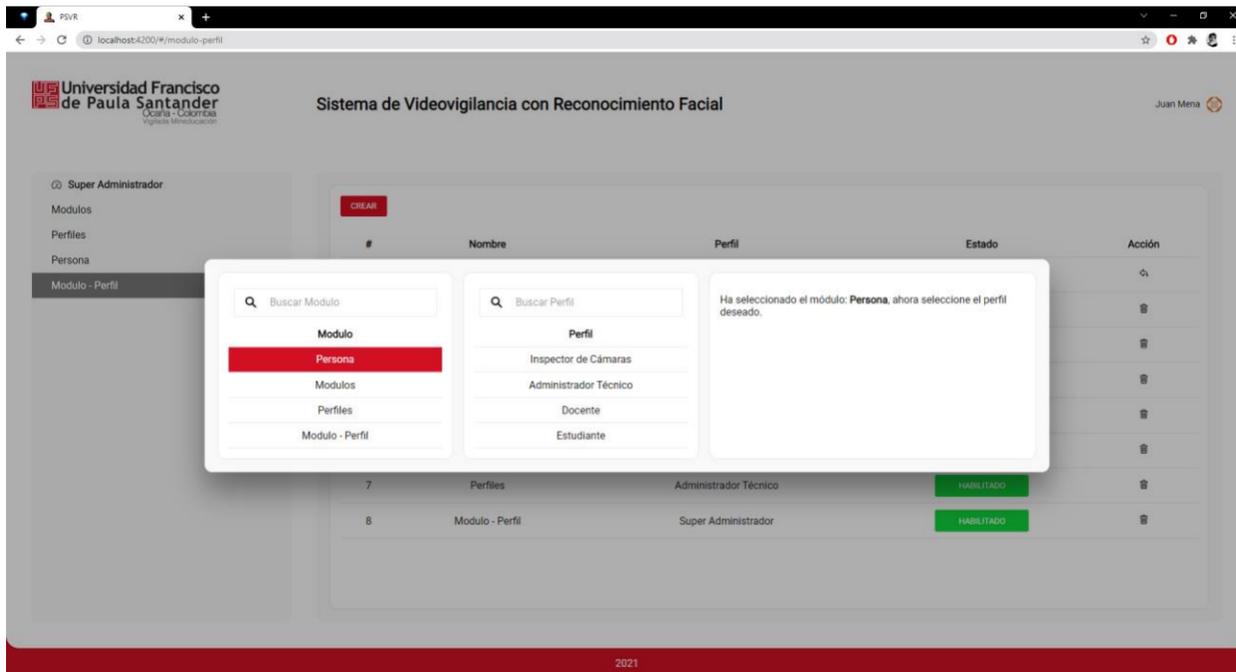


Nota. Realizado por los autores.

- Segmento de Perfiles:** Segmento con un buscador de perfiles y un listado de perfiles registrado en el sistema; el administrador podía realizar la búsqueda del perfil o clicar directamente sobre el listado de perfiles, para seleccionar la opción de perfil que deseaba asociar con el módulo, una vez realizada la acción el perfil seleccionado se resaltaba en color rojo.

Figura 134

Interfaz de selección del módulo

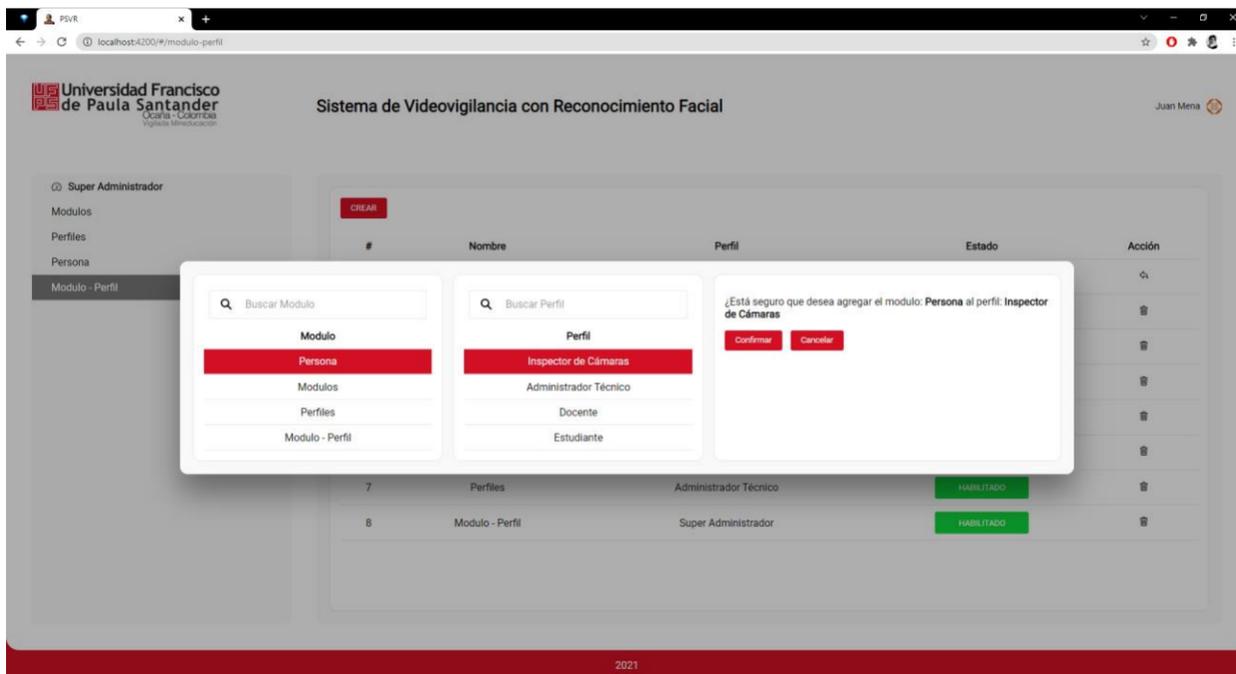


Nota. Realizado por los autores.

- Segmento de Seguimiento y Confirmación:** Segmento que presentaba al usuario los mensajes de validación de acuerdo con las interacciones realizadas durante el proceso, a su vez, cuando el proceso de selección finalizaba, se desplegaba un mensaje de confirmación de la asociación entre el módulo y el perfil, validado por medio de los botones “Confirmar” o “Cancelar”, afirmando que la asociación era correcta o descartando las opciones, permitiendo volver a realizar la selección de asociación desde el principio.

Figura 135

Interfaz de selección del módulo, el perfil y mensaje de confirmación



Nota. Realizado por los autores.

5.2.3.3 Evaluación del prototipo

En la tercera iteración se logró la construcción completa del CRUD donde los diferentes perfiles podían interactuar directamente con el sistema en todas sus funciones, pero donde los roles con mayor categoría podían realizar todos los procesos de gestión de los registros.

El prototipo permitía la ejecución de todas las funciones para el reconocimiento facial y el registro de las personas, a pesar de esto, se presentaban inconsistencias en la transferencia de

las imágenes de vídeo y caídas del servicio constantemente. De igual forma, se presentaban solapamiento de los fotogramas, presentando latencias en el despliegue de las imágenes de vídeo y las etiquetas representativas de la detección del rostro y el reconocimiento facial.

También, no existían mensajes de retroalimentación durante el proceso de entrenamiento del sistema, por lo que el usuario no podía percatarse del proceso mismo del entrenamiento, dejando en duda si se había realizado o no.

Es importante resaltar que durante el desarrollo de la tercera iteración se descartaron interfaces que se habían planteado en los mockups de la segunda iteración, debido a que no presentaban mayor significancia en la construcción del prototipo.

5.2.3.3.1 Nivel de satisfacción de los requerimientos. Durante la tercera iteración no se plantearon nuevos requerimientos en el sistema y se mantuvieron los mismos niveles de satisfacción de las iteraciones anteriores, a pesar de esto, se plantearon las siguientes correcciones:

- Revisar retroalimentación de acuerdo con las acciones realizadas por el usuario dentro del sistema.
- Adicionar mensajes de validación y confirmación de los procesos al ejecutar determinada función.
- Reestructurar carga de fotogramas de vídeo, evitando solapamiento y latencias en la transmisión.

- Ajustar mínimamente detalles de interfaz de usuario.

5.2.3.3.2 Refinamiento del prototipo. Con base en las correcciones planteadas y la evaluación del prototipo se estimó para la cuarta iteración, la corrección de errores de solapamiento y latencias, la adición de mensajes de retroalimentación y validación, y la corrección de errores de interfaz.

De igual forma se estimó la realización de pruebas de despliegue de múltiples clientes en un solo servicio de transmisión, con la intención de evaluar el comportamiento del sistema sobre una carga alta de sesiones.

5.2.4 Iteración 4

5.2.4.1 Requerimientos

Durante el desarrollo de la cuarta iteración no se realizaron modificaciones a los requerimientos del sistema, por tanto, a nivel de hardware como de software se mantuvieron las mismas herramientas.

5.2.4.2 Diseño y construcción

5.2.4.2.1 Inconvenientes y corrección de errores. Como se mencionó en la tercera iteración, durante el desarrollo se presentaron varios inconvenientes y errores de programación que requerían ser tratados para la optimización del prototipo.

Cuando se requería transmitir las imágenes de vídeo de forma independiente en la sección de monitoreo y la ventana emergente para el registro de las personas, la transmisión presentaba problemas de latencias que ocasionaban la repetición de fotogramas en la emisión de vídeo en tiempo real al ejecutar múltiples peticiones en una misma sesión, instanciando un mismo elemento de vídeo por la liberación inoportuna del recurso, razón por la que se implementó una nueva suscripción a las imágenes de video desde Angular.

Durante las pruebas realizadas para el reconocimiento facial, se encontraron varias inconsistencias, como la incompatibilidad entre los rasgos biométricos de las personas registradas y la visualización de los nombres al efectuarse el reconocimiento facial, así mismo, la ineficiencia en la detección de rostros al no identificar a las personas previamente registradas en la base de datos, debido a esto, se procedió con el desarrollo de una serie de funciones que permitieran capturar un mayor volumen de imágenes, mejorando el proceso de reconocimiento. Por otra parte, para corregir los inconvenientes de retroalimentación del sistema en la captura de los rasgos biométricos, se utilizaron herramientas como IONIC, considerando que ofrecía una gran variedad de estilos fáciles de utilizar con líneas simples de código.

5.2.4.2.2 Interfaces de Usuario.

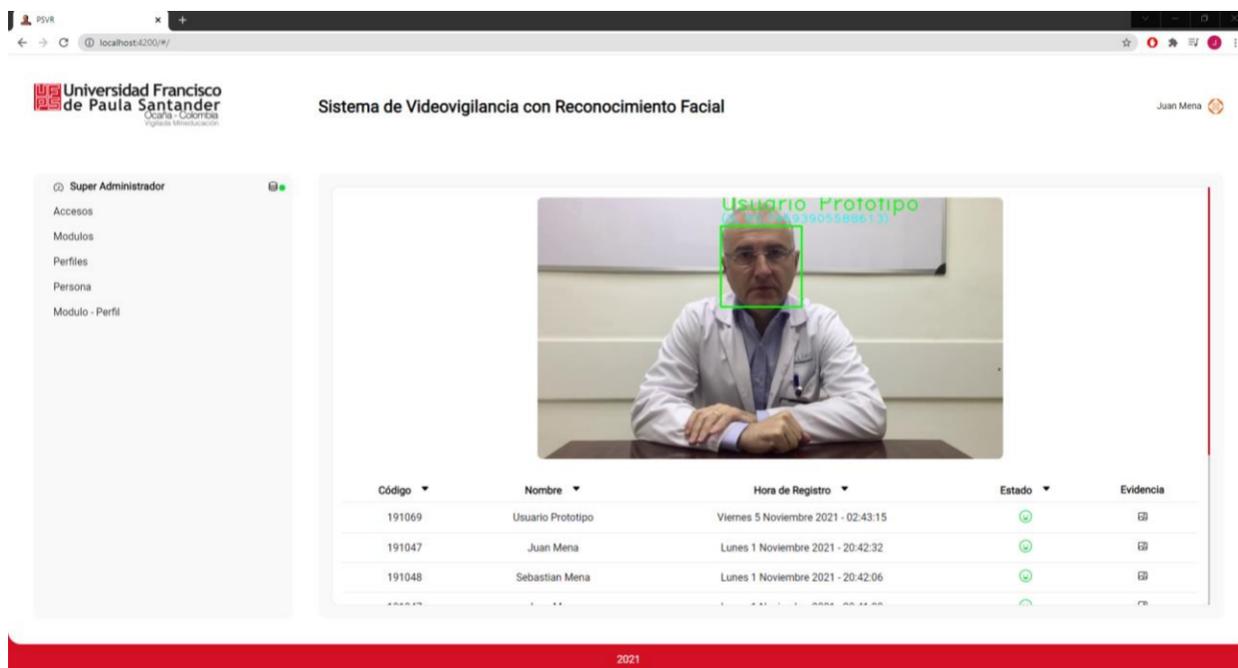
Interfaz de monitoreo. Cuando el usuario ingresaba al sistema, en la interfaz de monitoreo no se lograba visualizar ningún tipo de indicador diferente a los ya mencionados en la tercera iteración, por lo que en la cuarta iteración se realizaron varias modificaciones.

Dentro de las modificaciones realizadas, la de mayor relevancia fue la adición de un icono de retroalimentación que permitiera identificar si el servidor estaba activo o inactivo, ya que durante el desarrollo de la tercera iteración se presentaron múltiples inconsistencias con el servicio de UVICORN, generando el desconocimiento del estado del servicio.

Para identificar si el servicio estaba activo, se mostraba al administrador un icono relacionado con el servidor y un círculo de color verde () , de lo contrario si estaba inactivo, se mostraba el mismo icono, pero con el círculo en color rojo ().

Figura 136

Registro de personas conocidas



Nota. Realizado por los autores.

De igual forma, dentro de la interfaz de monitoreo se centró la imagen de vídeo de la cámara de videovigilancia, redondeando los bordes para que fuera más agradable a la vista, y se construyó una tabla con los datos de las personas que iban siendo registradas por el sistema, siendo estos: “Código”, “Nombre”, “Hora de Registro”, “Estado” y “Evidencia”.

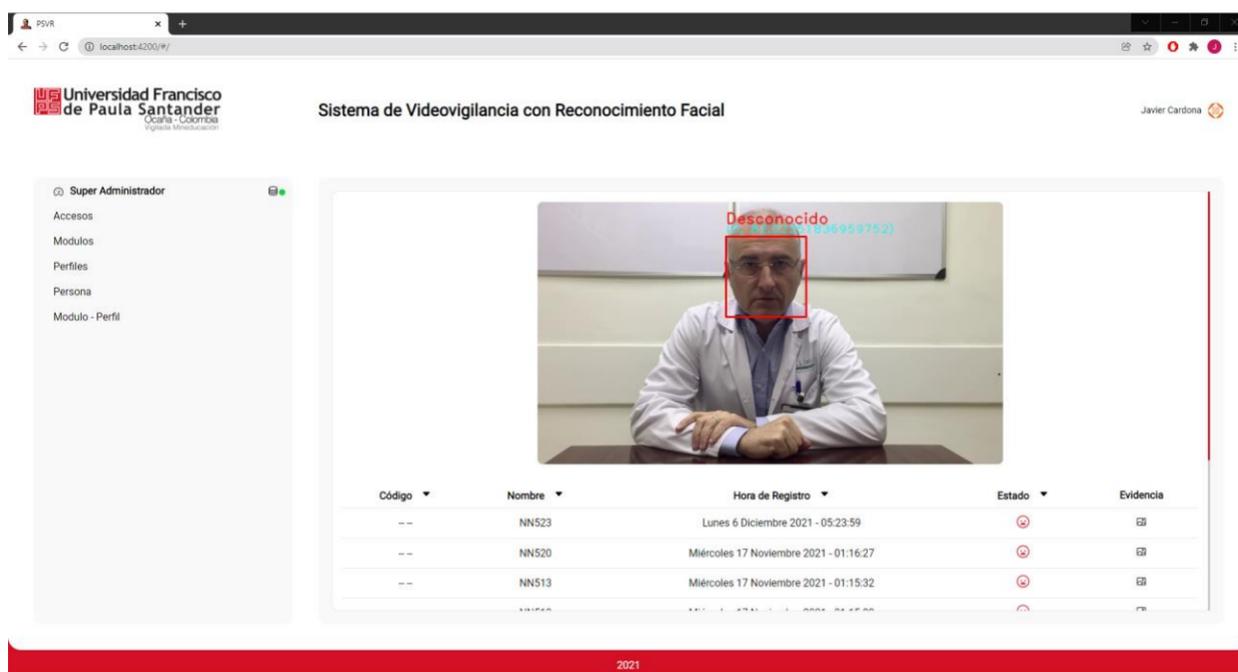
Como se planteó en el “Mockup de la interfaz de inspección de personas conocidas y desconocidas”, el sistema contaría con una tabla de registro para almacenar a todas las personas que fueran capturadas a través de la cámara de videovigilancia.

Durante la construcción del prototipo se replanteó el diseño, pero se desarrolló bajo el mismo concepto, aunque para reconocer el estado del registro, se utilizaron iconos representativos del rostro, siendo así como se seleccionó un icono de una cara feliz de color

verde para las personas que fueran conocidas (😊) y un icono de una cara triste de color rojo para las desconocidas (😞). De igual forma, por medio de un icono cliqueable de una imagen (🖼️), el administrador podía visualizar la evidencia del registro que fue almacenado.

Figura 137

Registro de personas desconocidas



The screenshot displays a web application titled 'Sistema de Videovigilancia con Reconocimiento Facial' from Universidad Francisco de Paula Santander. The interface includes a sidebar menu with options like 'Super Administrador', 'Accesos', 'Modulos', 'Perfiles', 'Persona', and 'Modulo - Perfil'. The main content area features a video feed of a man in a white lab coat, with a red bounding box around his face and the word 'Desconocido' above it. Below the video is a table with the following data:

Código	Nombre	Hora de Registro	Estado	Evidencia
--	NN523	Lunes 6 Diciembre 2021 - 05:23:59	😞	🖼️
--	NN520	Miércoles 17 Noviembre 2021 - 01:16:27	😞	🖼️
--	NN513	Miércoles 17 Noviembre 2021 - 01:15:32	😞	🖼️

Nota. Realizado por los autores.

Figura 138

Evidencia del registro de las personas capturadas



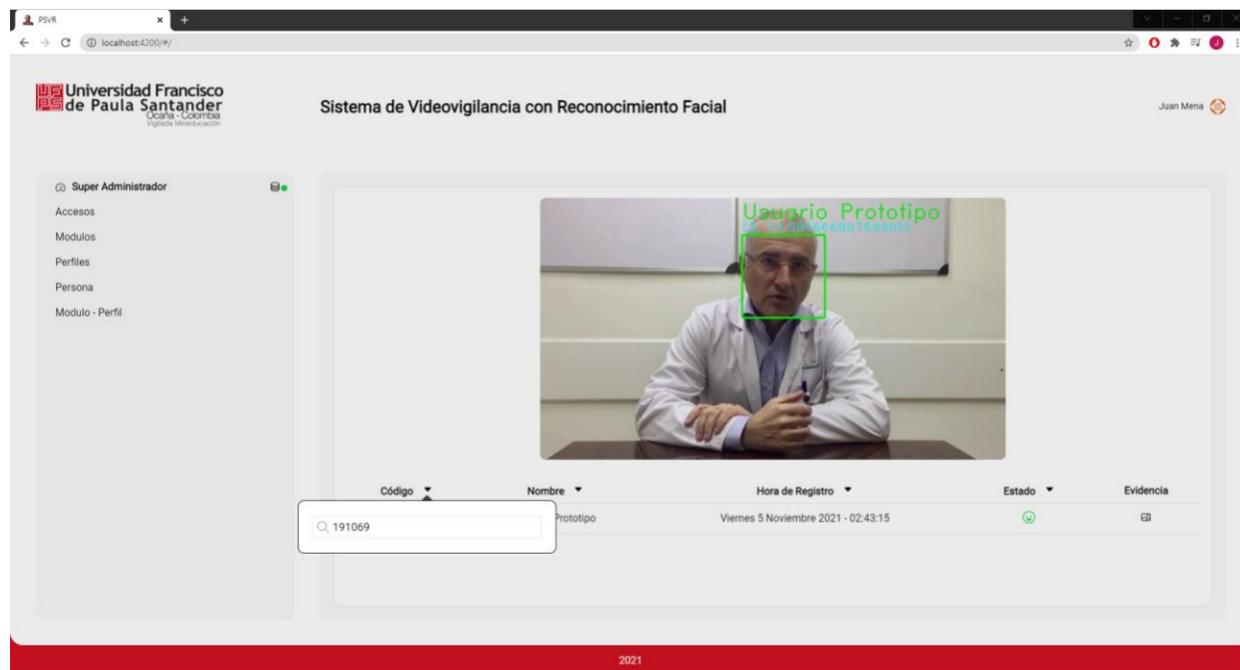
Nota. Realizado por los autores.

Filtros de la interfaz de monitoreo. Como se mencionó anteriormente, para la interfaz de monitoreo se contruyó una tabla de registro de las personas que iban siendo capturadas por el sistema, dichos registros podían ser filtrados de acuerdo a un campo en específico que le permitía al administrador reducir la cantidad de filas de la tabla al clickear en el icono de la flecha (▼) situado al costado derecho del título del filtro, indicando el criterio de búsqueda.

- **Filtro por Código:** Este filtro desplegaba una ventana emergente con una caja de texto para digitar caracteres numericos, permitiendo buscar a la persona dentro de los registros de acuerdo con el código asociado, mostrando las coincidencias en el orden de entrada. Por otra parte, al considerar que las personas desconocidas no contaban con un código asignado, el filtro no ejercía ninguna funcionalidad.

Figura 139

Filtro de personas reconocidas - Código



Nota. Realizado por los autores.

- **Filtro por Nombre:** Este filtro desplegaba una ventana emergente con una caja de texto para digitar caracteres alfabéticos, permitiendo buscar las coincidencias de las personas conocidas o desconocidas de acuerdo con el nombre almacenado o el nombre designado por el sistema, mostrando los resultados según el orden de entrada.

Figura 140

Filtro de personas reconocidas – Nombre de usuario

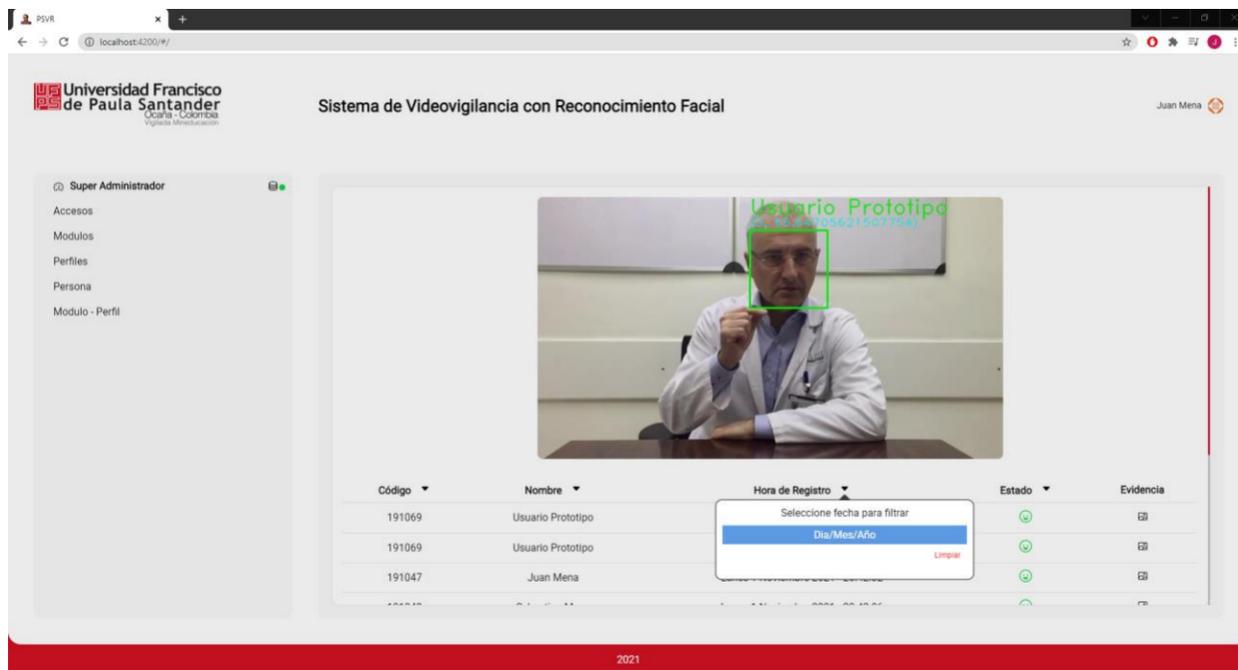


Nota. Realizado por los autores.

- Filtro por Hora de Registro:** Este filtro desplegaba un ventana emergente con el mensaje “Seleccione fecha para filtrar”, la opción “Día/Mes/Año” y la opción “Limpiar”, al clicar en la opción “Día/Mes/Año”, se desplegaba otra ventana emergente con las opciones “Cancelar”, “Seleccionar” y las opciones de modificación de la fecha (día, mes y año).

Figura 141

Filtro de personas reconocidas – Hora de Registro

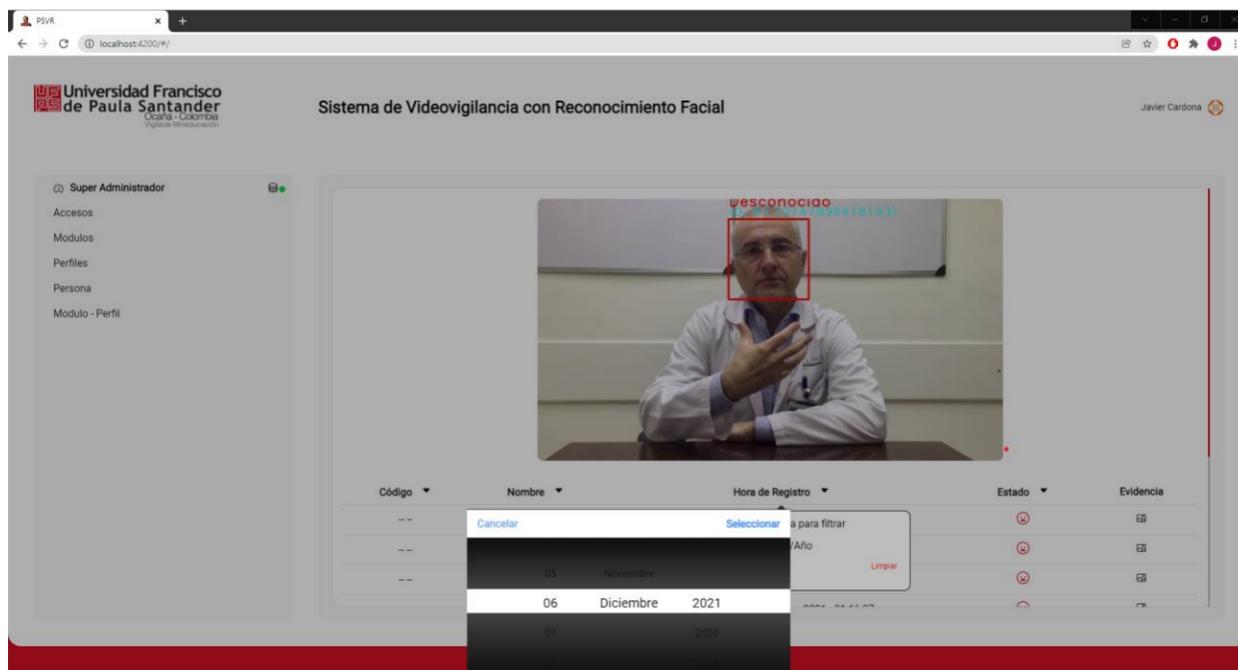


Nota. Realizado por los autores.

Si el usuario cliqueaba en la opción “Cancelar”, la segunda ventana emergente se cerraba, sin embargo, si cliqueaba en la opción “Seleccionar”, se configuraba el filtro de acuerdo con la fecha estipulada al realizarse la modificación, logrando visualizar los registros de la tabla según el orden con que fueron almacenado por la fecha designada. Por otra parte, si el usuario cliqueaba en la opción “Limpiar” de la primera ventana emergente, se reiniciaba la configuración y el filtro tomaba su valor por defecto.

Figura 142

Selección de la fecha



Nota. Realizado por los autores.

- **Filtro por Estado:** Este filtro permitía buscar los registros de la tabla de acuerdo con el estado de reconocimiento, desplegando una ventana emergente con el mensaje “Seleccione un estado para filtrar” y las opciones “Conocido”, “Desconocido” y “Limpiar”.

Si el administrador cliqueaba en la opción “Conocido”, se visualizaban en la tabla de registro todas las personas identificadas por el sistema, ó, en el caso contrario, si cliqueaba en la opción “Desconocido”, se filtraban todas las personas no identificadas, igualmente, si cliqueaba en la opción “Limpiar”, se reiniciaba la configuración del filtro.

Figura 143

Filtro de personas reconocidas – Estado conocido

Universidad Francisco de Paula Santander
VICARIA - LOGOTIPO
Vigilancia Minimization

Sistema de Videovigilancia con Reconocimiento Facial

Juan Mena

Super Administrador

- Accesos
- Modulos
- Perfiles
- Persona
- Modulo - Perfil

Usuario Prototipo
(191069, 2021-11-05 02:45:17)

Código	Nombre	Hora de Registro	Estado	Evidencia
191069	Usuario Prototipo	Viernes 5 Noviembre 2021 - 02:45:17	Conocido	
191069	Usuario Prototipo	Viernes 5 Noviembre 2021 - 02:44:16	Desconocido	
191069	Usuario Prototipo	Viernes 5 Noviembre 2021 - 02:43:15		

2021

Nota. Realizado por los autores.

Figura 144

Filtro de personas reconocidas – Estado desconocido

Universidad Francisco de Paula Santander
VICARIA - LOGOTIPO
Vigilancia Minimization

Sistema de Videovigilancia con Reconocimiento Facial

Juan Mena

Super Administrador

- Accesos
- Modulos
- Perfiles
- Persona
- Modulo - Perfil

Usuario Prototipo
(191069, 2021-10-31 18:22:03)

Código	Nombre	Hora de Registro	Estado	Evidencia
--	NN00131	Domingo 31 Octubre 2021 - 18:22:03	Desconocido	

2021

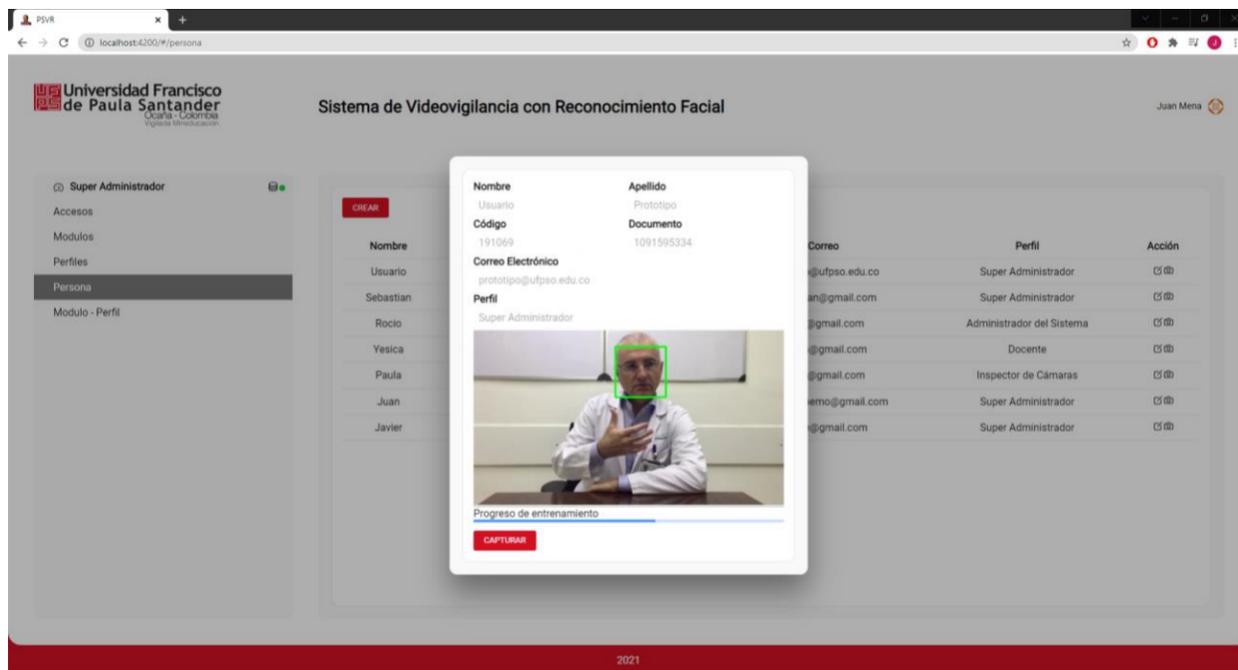
Nota. Realizado por los autores.

Interfaz de personas. Como se mencionó anteriormente en el desarrollo de las interfaces dentro de la tercera iteración, una vez el administrador desplegara la ventana emergente para capturar los rasgos biométricos de la persona y cliqueara en el botón “Capturar”, el sistema iniciaría el proceso de matriculación, pero, durante esta etapa del procedimiento no se contaba con un indicador que retroalimentara el progreso del entrenamiento, por este motivo, se implementó una barra de carga.

Para el progreso de carga de la barra de entrenamiento se estructuró un bloque de código dentro de las funciones de registro de los rasgos biométricos, realizando un cálculo porcentual referente a la cantidad de imágenes que iban siendo capturadas al detectar el rostro de la persona, debido a que, al no detectarse ningún rostro, la barra de carga no aumentaría y permanecería de color azul.

Figura 145

Interfaz de retroalimentación del entrenamiento del modelo – Progreso de entrenamiento



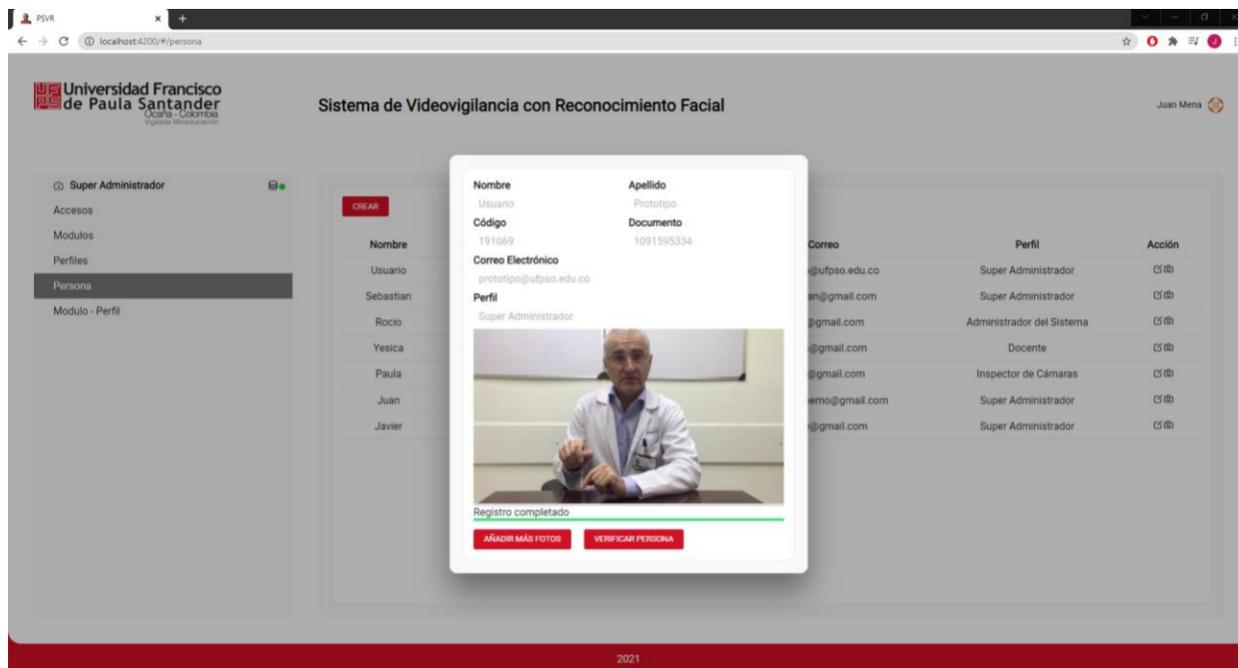
Nota. En la imagen se aprecia la barra de carga que valida que se está entrenando el sistema.

Realizado por los autores.

Una vez completado el registro de acuerdo con la cantidad de fotos establecidas para la extracción del rostro y el entrenamiento del modelo, se mostraba al usuario el mensaje de validación “Registro Completado” junto a la barra de carga en color verde, confirmando la efectividad del proceso y la habilitación de los botones “Añadir Fotos” y “Verificar Persona”.

Figura 146

Interfaz de retroalimentación del entrenamiento del modelo – Registro completado

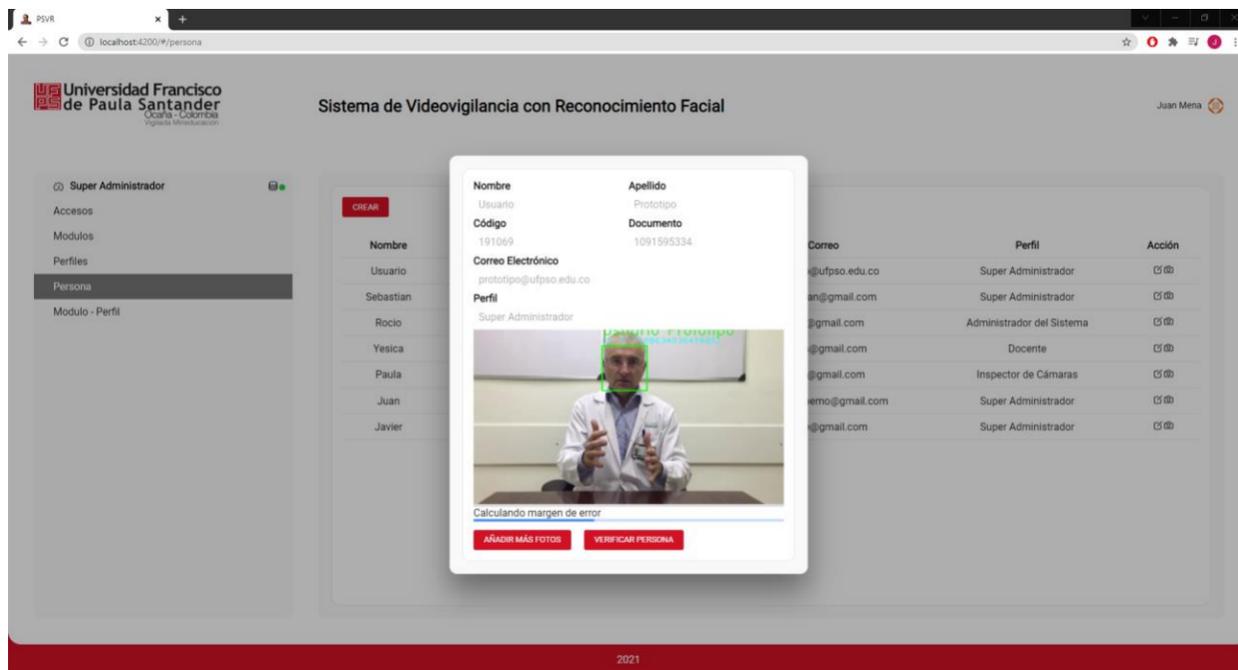


Nota. Realizado por los autores.

Quando el administrador cliqueara en el botón “Añadir Fotos”, el proceso de matriculación volvería a efectuarse y se reentrenaría el modelo, añadiendo una serie de fotos extras del rostro de la persona en el conjunto de datos previamente almacenado en la primera ronda, esta función había sido diseñada para efectuarse si el proceso de verificación de la persona fallaba al cliquear en el botón “Verificar Persona”.

Figura 147

Interfaz de retroalimentación del entrenamiento del modelo – Cálculo de margen de error

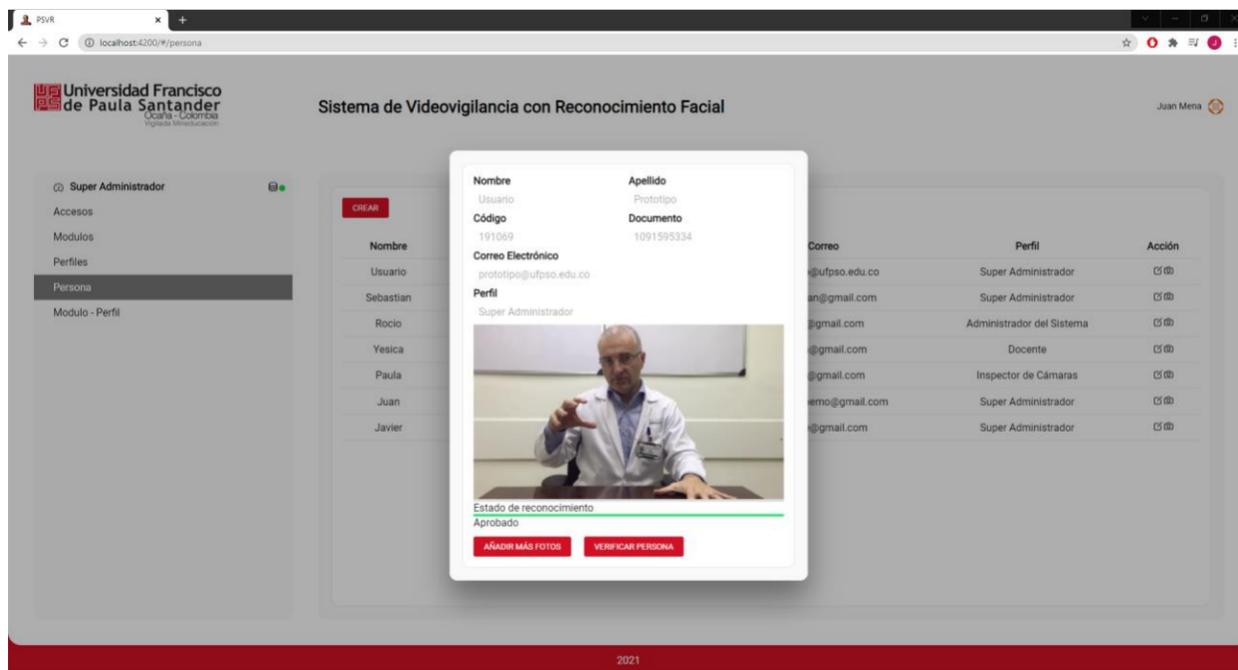


Nota. Realizado por los autores.

Una vez se iniciaba al proceso de verificación del modelo N de personas registradas, el sistema comenzaba a realizar el cálculo del margen de error, mostrando el mensaje “Calculando margen de error” y la barra de carga en color azul, este proceso ejecutaba un algoritmo en relación al nivel de reconocimiento de la persona con sus rasgos biométricos, siendo así, que si se reconocía completamente a la persona durante la secuencia, el sistema aprobaría la asociación biométrica y la barra de carga cambiaría a color verde, en caso que se dieran intervalos de reconocimiento y desconocimiento de la persona, el algoritmo determinaría el porcentaje de validación respecto a un margen de error estipulado, si se superaba el umbral porcentual de error, la persona sería aprobada en su asociación biométrica, de lo contrario, el sistema la desaprobaba; siendo el mismo caso si no se reconociera a la persona en ninguna parte de la secuencia y la barra de carga cambiaría a color rojo.

Figura 148

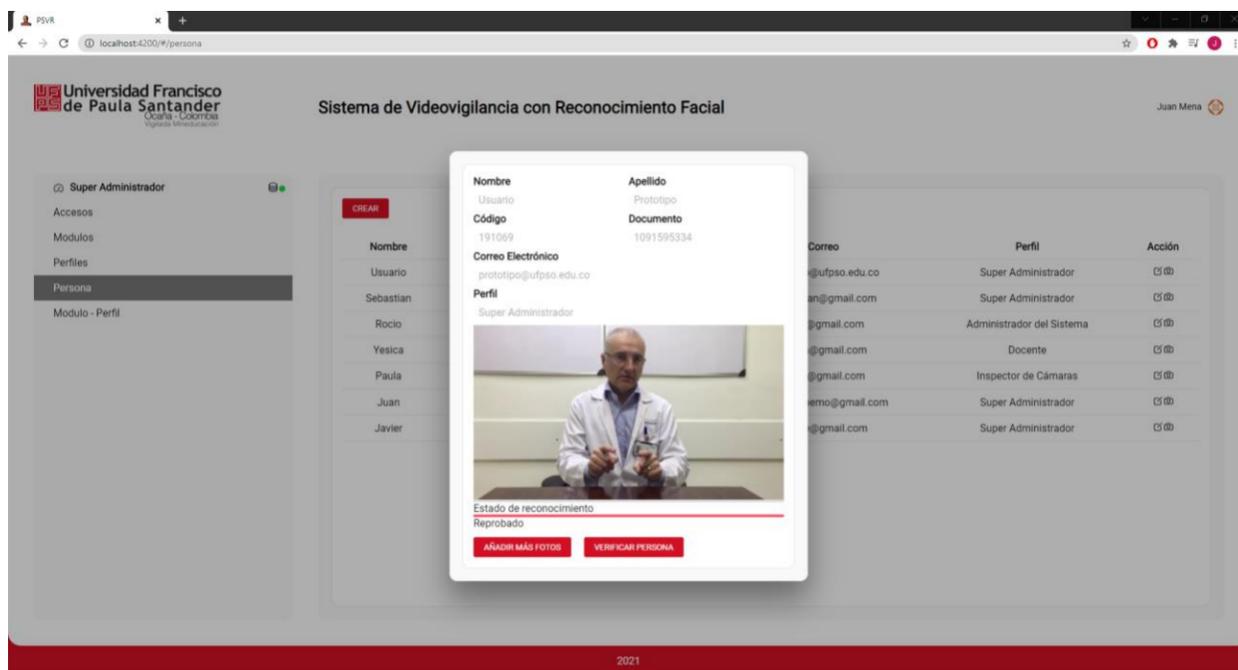
Interfaz de retroalimentación del entrenamiento del modelo – Registro aprobado



Nota. Realizado por los autores.

Figura 149

Interfaz de retroalimentación del entrenamiento del modelo – Registro reprobado



Nota. Realizado por los autores.

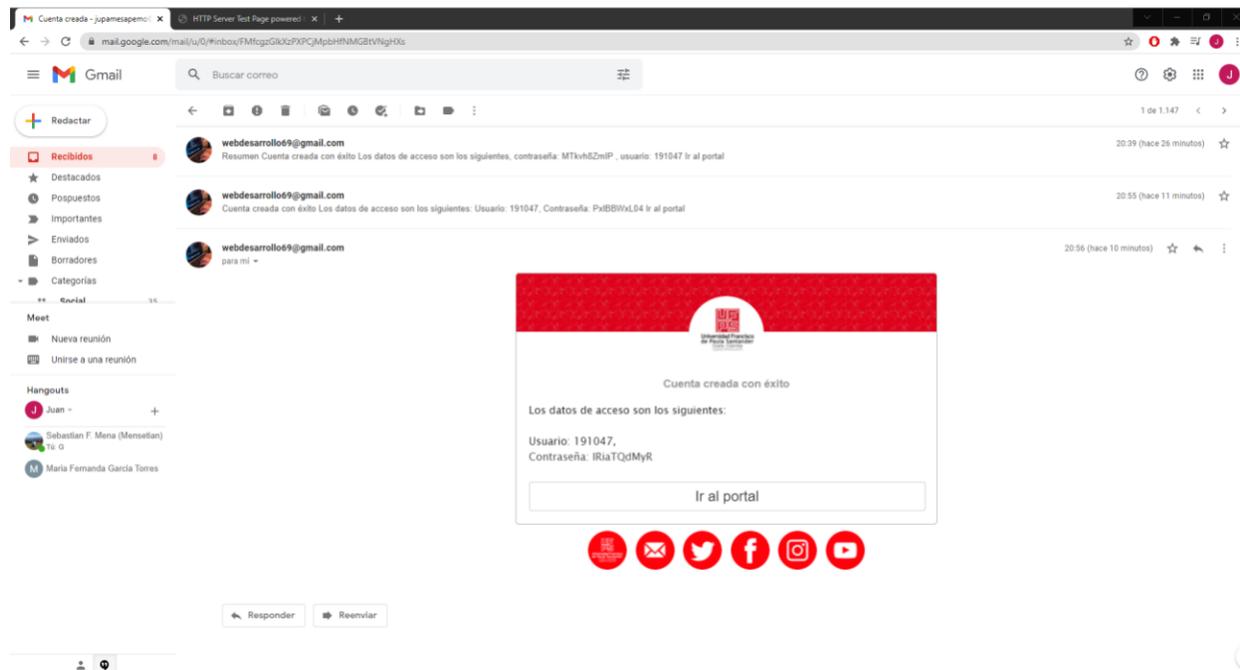
Por consiguiente, si no se aprobaba el registro biométrico de la persona en el proceso de verificación, la persona encargada de registrar a los usuarios debería realizar de nuevo el proceso de matriculación y verificación, hasta que el usuario fuera aprobado.

5.2.4.2.3 Vista del correo electrónico.

Crear cuenta. Una vez finalizado el proceso de asignación de permisos, se les notificaba a los usuarios la creación exitosa de la cuenta a través de un correo electrónico con una plantilla institucional que contenía las credenciales de acceso al sistema (usuario y contraseña) y las opciones de “Ir al portal” para redireccionarse hacia la página de inicio de sesión del prototipo de reconocimiento facial o las redes sociales de la universidad.

Figura 150

Interfaz del correo electrónico – Creación de cuenta

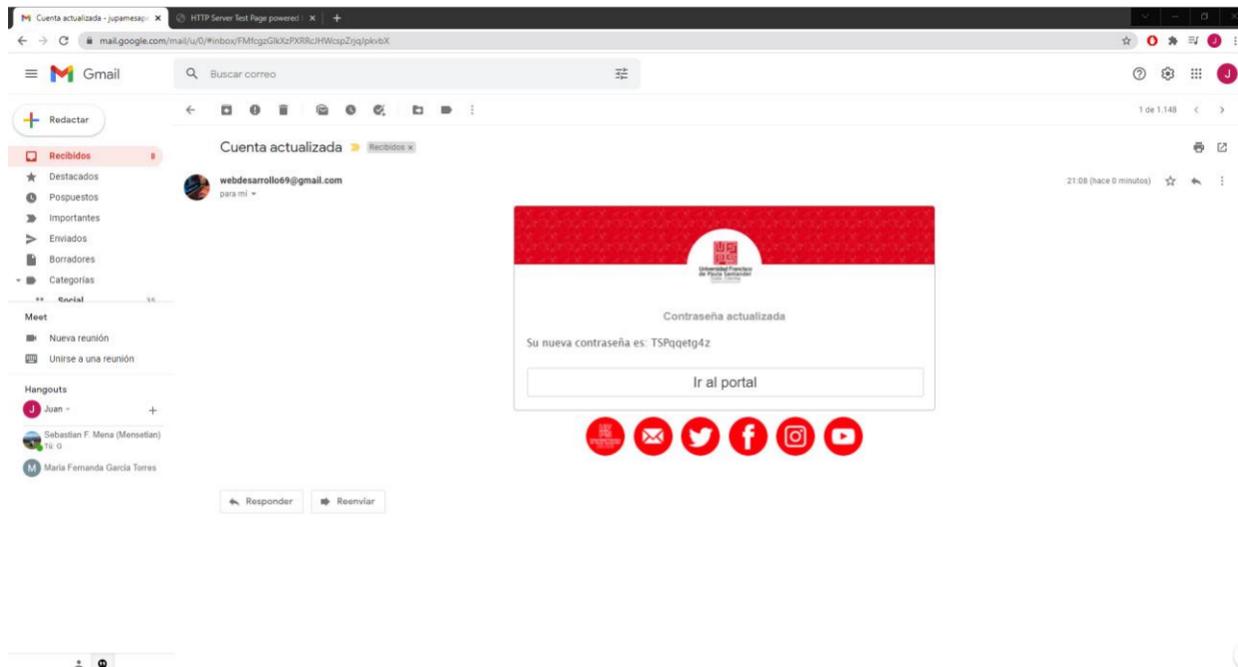


Nota. Realizado por los autores.

Actualizar contraseña. En caso de que un usuario con acceso al sistema olvidara su contraseña, el administrador tenía la posibilidad de reasignarle una nueva, notificándole a través de un correo electrónico con las mismas características mencionadas anteriormente la nueva contraseña asignada.

Figura 151

Interfaz del correo electrónico – Actualización de contraseña

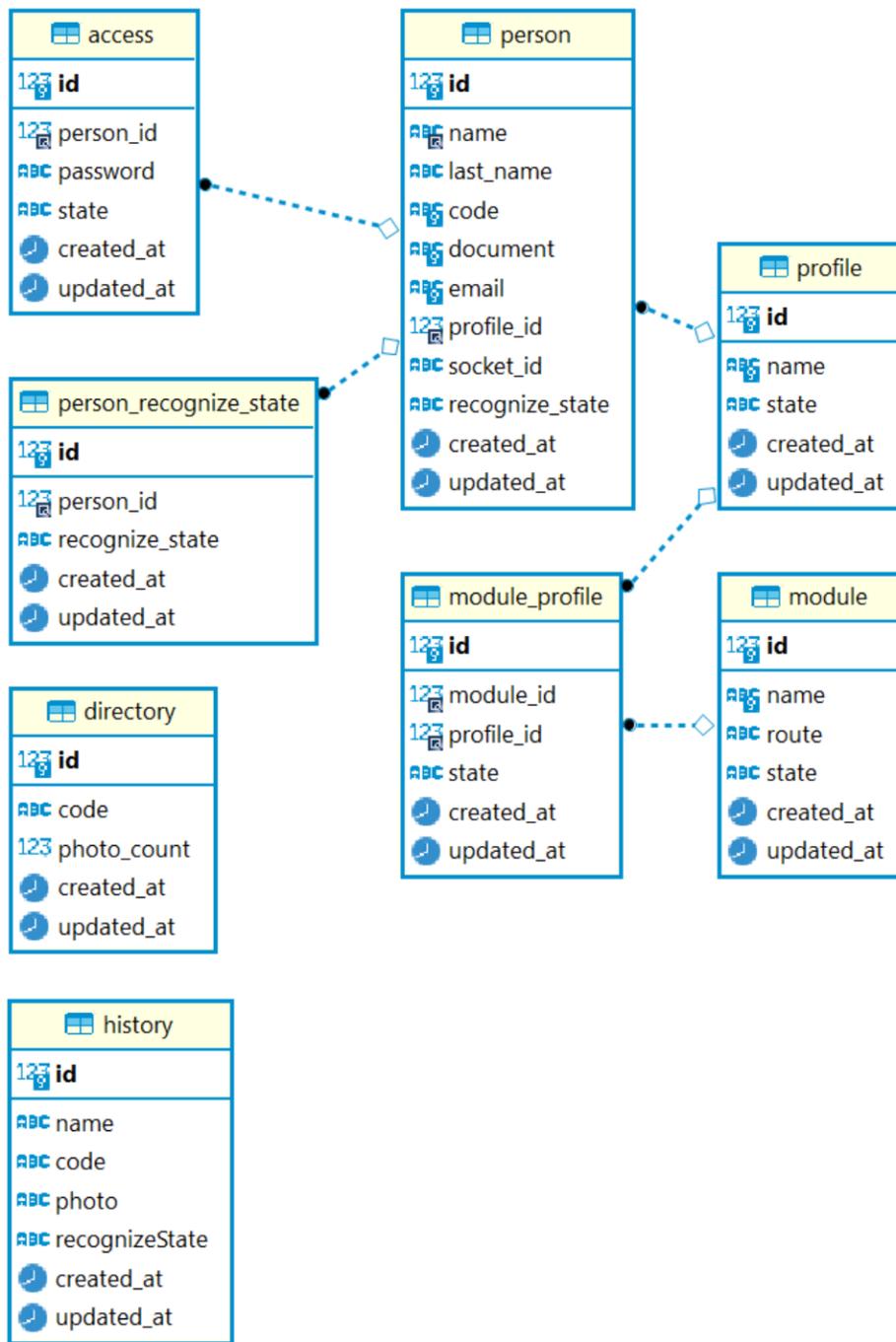


Nota. Realizado por los autores.

5.2.4.2.4 Modelo relacional del sistema - Modificado.

Figura 152

Modelo relacional del sistema en la iteración 4



Nota. Realizado por los autores.

5.2.4.2.5 Detalle de modificación de las tablas del modelo relacional.

Tabla 42*Detalle de las tablas del modelo relacional - Iteración 4*

Detalle de las Tablas del Modelo Relacional		
Nombre de la Tabla	Descripción (Modificación)	Relación
“history”	Permite gestionar la información del historial de registro, almacenando un identificador, el nombre de la persona conocida o desconocida, el código de la persona conocida, la url de la evidencia capturada, el estado reconocimiento y la fecha de creación del registro.	No posee relación.

Nota. Realizado por los autores.

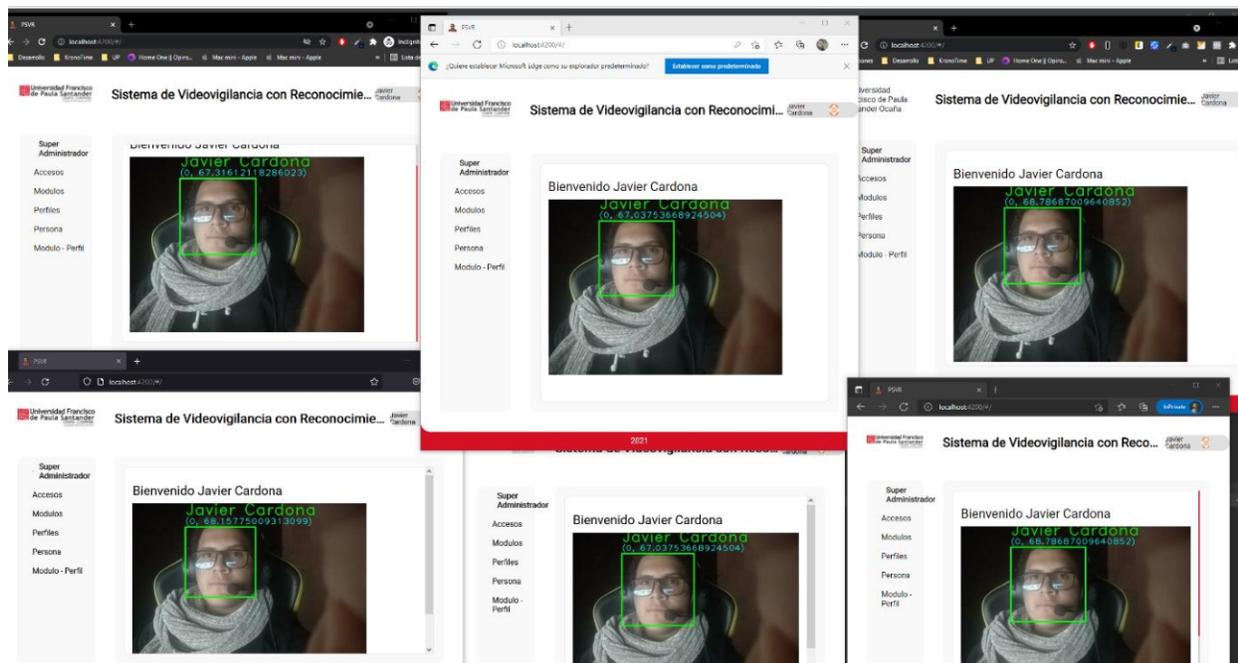
Las modificaciones realizadas a la base datos en la cuarta iteración, tienen como finalidad llevar un control de todas las personas capturadas en el entorno de prueba durante la fase de implementación, para su posterior visualización en la interfaz de monitoreo del sistema.

5.2.4.2.6 Pruebas de rendimiento y verificación de modificaciones.

Prueba de despliegue de múltiples sesiones de transmisión. Con el fin de verificar la estabilidad del servicio, se realizó una prueba de concurrencia, validando que se pudieran desplegar múltiples sesiones en distintos navegadores y pestañas donde se llevara a cabo el reconocimiento facial, considerando que, en iteraciones anteriores, el sistema colapsaba y se generaba la baja del servicio o el retraso de la transmisión.

Figura 153

Interfaz de despliegue de múltiples sesiones de transmisión



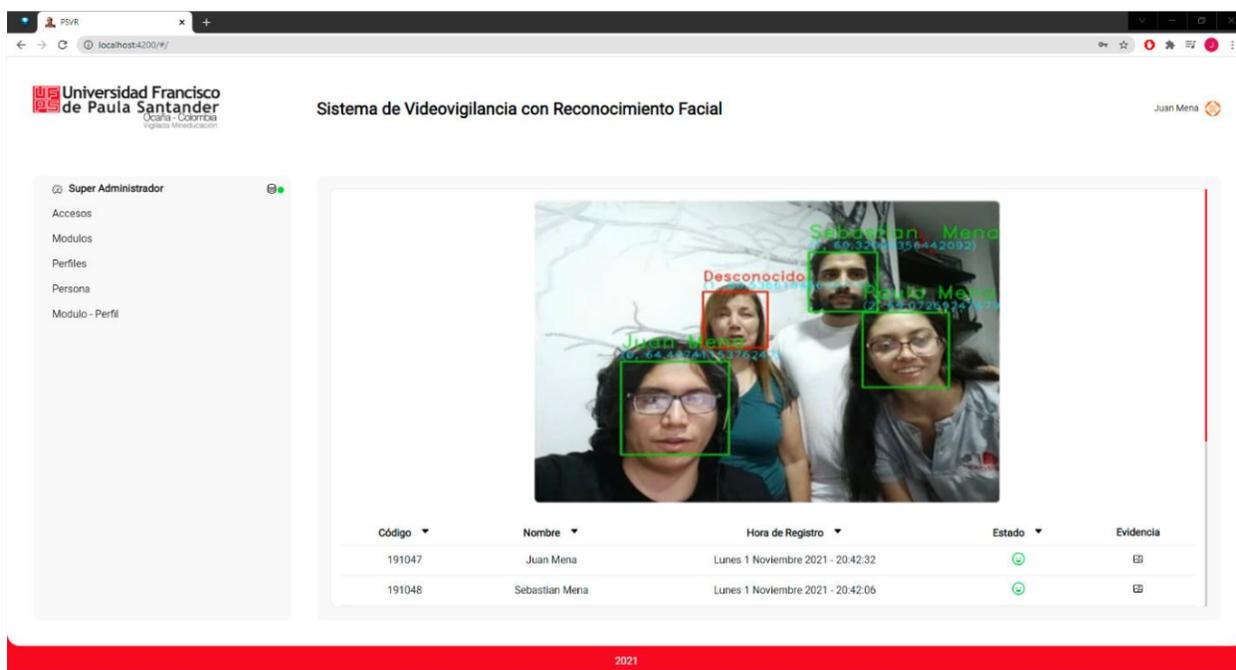
Nota. Realizado por los autores.

Prueba de reconocimiento facial simultaneo con varias personas registradas. De igual forma, se realizó una prueba de verificación de reconocimiento facial con varias personas que se encontraban registradas en el sistema, con el objetivo de validar el correcto funcionamiento del clasificador y llevar un historial de registro de las personas reconocidas en la interfaz de monitoreo. Al realizar esta prueba y captura de evidencia, el sistema no contaba aún con un control de las personas desconocidas, por tal motivo y como se aprecia en la **Figura 78**, la persona que aparece con la etiqueta “Desconocido”, no se encontraba registrada en la tabla de control, sin embargo, posteriormente se codificó un función para registrar personas desconocidas.

Por consiguiente, con la intención de llevar un control adecuado de las personas desconocidas, se programó una estructura de código capaz de almacenar temporalmente a la persona desconocida e identificar si previamente había sido almacenada, con el objetivo evitar redundancia de información en cuanto al registro de una misma persona se refiere.

Figura 154

Registro de múltiples personas detectadas



Código	Nombre	Hora de Registro	Estado	Evidencia
191047	Juan Mena	Lunes 1 Noviembre 2021 - 20:42:32	✓	📄
191048	Sebastian Mena	Lunes 1 Noviembre 2021 - 20:42:06	✓	📄

Nota. Realizado por los autores.

5.2.4.3 Evaluación del prototipo

En la cuarta iteración se lograron llevar a cabo las correcciones planteadas de solapamiento y latencias en las imágenes de vídeo transmitidas, además se mejoraron todos los

mensajes de retroalimentación de acuerdo con el proceso en ejecución para efectuar el reconocimiento facial.

Por otra parte, se mejoraron los mensajes de correo electrónico recibidos por los usuarios con acceso al sistema cuando se les asignaban credenciales o cuando necesitaban recuperar la contraseña, también, se programaron todos los filtros planteados para analizar los registros almacenados durante el reconocimiento facial y se hicieron pruebas de funcionamiento para validar el reconocimiento de múltiples personas dentro de la imagen de vídeo en tiempo real y el comportamiento del sistema sobre una carga alta de sesiones.

5.2.4.4 Nivel de satisfacción de los requerimientos. Durante la cuarta iteración no se plantearon nuevos requerimientos en el sistema y se mantuvieron los mismos niveles de satisfacción de las iteraciones anteriores.

5.2.4.4.1 Refinamiento del prototipo. En este punto el prototipo era completamente funcional, necesitaba algunos ajustes mínimos, pero podía ser implementado en la universidad, siendo así que para la quinta iteración se estipuló la implementación del sistema para evaluar su comportamiento.

5.3 Implementación del sistema

5.3.1 Iteración 5

5.3.1.1 Requerimientos

Para llevar a cabo la implementación del prototipo en la quinta iteración, se realizó la reasignación de la ubicación del prototipo, siendo así como dentro de los requerimientos, el sitio de ubicación pasaría de ser el “Edificio de Ingeniería”, a ser el “Laboratorio de Redes y Telecomunicaciones”, debido a inconvenientes de logística y practicidad. Además, teniendo en cuenta que es un prototipo experimental y está orientado no solo a velar por la seguridad e integridad del personal universitario, también está enfocado dentro del panorama investigativo.

5.3.1.2 Implementación

Para efectuar la implementación se hizo necesaria la comunicación con el personal universitario encargado de la gestión de los servidores, donde, se logró programar un bloque de código ejecutado en el servidor para levantar el servicio y prever caídas del mismo, de igual forma, se configuraron los servicios de hosting, dominios y bases de datos, tanto como el enrutamiento de la cámara de videovigilancia por medio de una IP estática dentro de la red del laboratorio.

Figura 155

Construcción del soporte para la cámara de videovigilancia

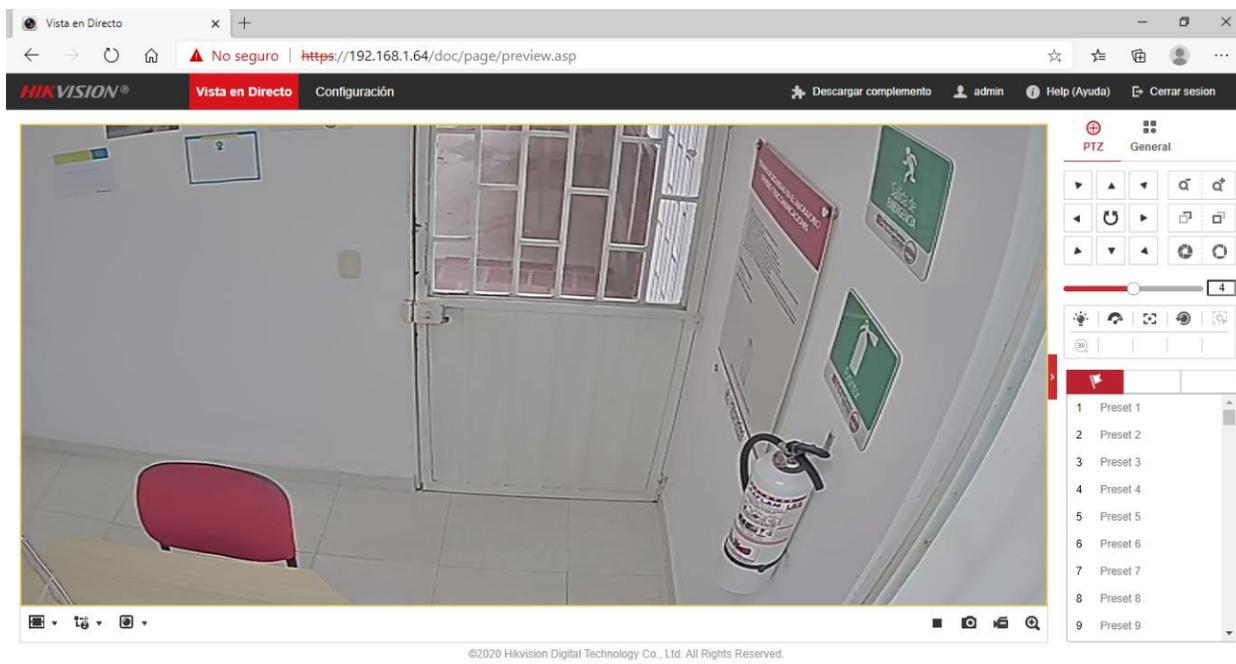


Nota. Realizado por los autores.

Por efectos de practicidad, se construyó una base en madera para posicionar la cámara sobre un soporte metálico dentro del laboratorio, este soporte tenía la ventaja de ser reposicionado según se necesitara, siguiendo la lógica del prototipo, este se ubicó apuntando hacia la puerta de entrada del laboratorio.

Figura 156

Prueba de visualización de la cámara de videovigilancia en el laboratorio de redes y telecomunicaciones



Nota. Realizado por los autores.

Figura 157

Base en madera de la cámara de videovigilancia, sujeto al soporte metálico



Nota. Realizado por los autores.

También, de acuerdo con el orden de los dispositivos instalados dentro del laboratorio de redes y telecomunicaciones, se decidió pintar la base de madera de color blanco.

Figura 158

Base de madera pintada de color blanco



Nota. Realizado por los autores.

5.3.1.3 Evaluación del prototipo

5.3.1.3.1 Nivel de satisfacción de los requerimientos. En la quinta iteración se realizó el mapeo de los requerimientos satisfechos, de acuerdo con los resultados establecidos por todas las iteraciones, teniendo en cuenta las modificaciones planteadas sobre la ubicación e implementación del prototipo en el laboratorio de redes y telecomunicaciones.

*Requerimientos iniciales.***Tabla 43***Nivel de satisfacción de los requerimientos iniciales - Iteración 1*

Nivel de Satisfacción de los Requerimientos Iniciales		
No.	Nivel de Satisfacción	Observación
RIS1	S	El sistema permitía la detección de rostros en tiempo real.
RIS2	S	El sistema permitía el almacenamiento de los rostros en ficheros organizados con 300 imágenes redimensionadas del rostro de la persona, según le correspondía.
RIS3	S	Se utilizó un cámara de videovigilancia conectada a través del puerto Ethernet y configurada con una IP estática.
RIS4	S	Se realizó la renderización de la imagen de vídeo.
RIS5	S	El sistema estuvo conectado las 24 horas del día a la red eléctrica.
RIS6	S	El sistema permitía reconocer y almacenar personas identificadas en una base de datos realizada en MySQL.
RIS7	S	El sistema permitía reconocer personas no identificadas que podían ser registradas dentro del sistema para su integración.
RIS8	MS	El prototipo se implementó en el Laboratorio de Redes y Telecomunicaciones.
RIS9	S	Se realizó la obtención del vídeo por medio de la cámara de videovigilancia.
RIS10	S	El sistema permitía permanentemente el procesamiento de las imágenes obtenidas por la cámara.
RIS11	MS	El sistema a pesar de no ser discreto, no interfería con las actividades de las personas que hacían uso del recinto.
RIS12	S	El sistema se implementó en un lugar estratégico.

Nivel de Satisfacción de los Requerimientos Iniciales		
No.	Nivel de Satisfacción	Observación
RIS13	S	La cámara de seguridad se ubicó a una altura idónea que permitiera la captura del rostro de las personas.

Nota. Realizado por los autores.

Requerimientos funcionales.

Tabla 44

Nivel de satisfacción de los requerimientos funcionales - Iteración 1

Nivel de Satisfacción de los Requerimientos Funcionales		
No.	Nivel de Satisfacción	Observación
RU1	MS	El sistema se implementó en el Laboratorio de Redes y Telecomunicaciones.
RU2	S	El sistema permitía la adquisición de la información para realizar el entrenamiento.
RU3	S	El sistema permitía el almacenamiento de los rostros de las personas en directorios estructurados y organizados, para la realización del entrenamiento del modelo.
RU4	S	El sistema permitía al administrador realizar las operaciones CRUD para gestionar los accesos al sistema (Nombre, Apellido, Código, Documento, Correo, Estado, Acción editar, Acción habilitar, Acción inhabilitar).
RU5	S	El sistema permitía al administrador realizar las operaciones CRUD para gestionar los módulos del sistema (Nombre, Ruta, Estado, Acción editar, Acción habilitar, Acción inhabilitar).
RU6	S	El sistema permitía al administrador realizar las operaciones CRUD para gestionar los perfiles del sistema (Nombre, Estado, Acción editar, Acción habilitar, Acción inhabilitar).

Nivel de Satisfacción de los Requerimientos Funcionales		
No.	Nivel de Satisfacción	Observación
RU7	S	El sistema permitía al administrador Realizar las operaciones CRUD para gestionar las personas que interactuarán con el sistema (Nombre, Apellido, Código, Documento, Correo, Perfil, Acción editar, Acción capturar rostro, Acción identificar persona).
RU8	S	El sistema permitía al administrador Realizar las operaciones CRUD para gestionar la asociación del módulo con el perfil del sistema (Nombre, Perfil, Estado, Acción habilitar, Acción inhabilitar, Buscar módulo, Buscar perfil).
RU9	S	El sistema permitía al administrador visualizar el streaming de las cámaras que identifican a las personas, sea monitoreo, registro de usuarios o verificación de rasgos biométricos de los usuarios en el sistema.
RU10	S	El sistema debe permitía el acceso a los usuarios por medio de credenciales (Código, Contraseña).
RU11	S	El sistema contaba con las condiciones para permitir identificar rostros sin ser supervisado.

Nota. Realizado por los autores.

Requerimientos no funcionales.

Tabla 45

Nivel de satisfacción de los requerimientos no funcionales - Iteración 1

Nivel de Satisfacción de los Requerimientos No Funcionales		
No.	Nivel de Satisfacción	Observación
RA1	MS	La cámara fue instalada sobre un soporte metálico y una base de madera, en una zona estratégica dentro del Laboratorio de Redes y Telecomunicaciones.

Nivel de Satisfacción de los Requerimientos No Funcionales		
No.	Nivel de Satisfacción	Observación
RA2	MS	Se realizó la conexión Ethernet, extendida a través del sistema de cableado estructurado de la universidad, desde el Laboratorio de Redes y Telecomunicaciones.
RA3	S	El servidor de la universidad está ubicado en un lugar adecuado con correcta ventilación.
RA4	S	El sistema se desarrolló mediante un CPU capaz de realizar cualquier tipo de procesamiento complejo.
RA5	MS	Se desconocían las características del servidor, por cuestiones de seguridad.
RA6	S	El sistema contaba con un PSU adecuada, para brindar el suministro de energía suficiente a todos los componentes del servidor.
RA7	S	El sistema contaba con una PSU con certificación, para garantizar la estabilidad del suministro de energía.
RA8	S	El sistema contaba con una tarjeta madre con conexión a Ethernet.
RA9	S	Se desconocían las características del servidor, por cuestiones de seguridad.
RA10	S	Se desconocían las características del servidor, por cuestiones de seguridad.
RA11	S	Se utilizó una cámara de seguridad de alta resolución.
RA12	S	Las imágenes eran procesadas eficientemente por el CPU.
RA13	S	El sistema contaba con almacenamiento suficiente para almacenar volúmenes considerables de información.
RA14	S	El CPU era compatible con cualquier tipo de arquitectura de SO disponibles.
RA15	S	El CPU cumplía con su función.

Nivel de Satisfacción de los Requerimientos No Funcionales		
No.	Nivel de Satisfacción	Observación
RA16	MS	Se desconocían las características del servidor, por cuestiones de seguridad.
RA17	MS	Se desconocían las características del servidor, por cuestiones de seguridad.
RA18	S	Se utilizó una cámara de seguridad con conexión Ethernet, configurada con una IP estática, dentro de la red LAN.
RA19	S	El sistema se desarrolló en Windows 10, pero podía ser implementado en cualquier SO, además, se desarrolló con herramientas de desarrollo de código abierto como Visual Studio Code, PyCharm, entre otras.
RA20	S	Se implementó el lenguaje de programación Python, por su nivel de compatibilidad con OpenCV y su capacidad de manejar tecnologías de inteligencia artificial de manera práctica.
RA21	S	El sistema permitía la ejecución de visión artificial en tiempo real.
RA22	S	El sistema permitía la ejecución de multiprocesos de forma eficaz.
RA23	S	El sistema permitía la utilización de los recursos de la GPU, para el entrenamiento de los modelos.
RA24	S	Se estructuró una base de datos realizada en MySQL.
RA25	S	Las herramientas de software eran compatibles entre sí para el uso de las bibliotecas de aprendizaje automático.
RA26	S	Las herramientas de software eran compatibles con tecnología de cómputo en paralelo.
RA27	S	El sistema permanecía conectado a la red eléctrica.
RA28	S	Se estipularon herramientas de desarrollo web que permitieran la construcción del prototipo.

Nivel de Satisfacción de los Requerimientos No Funcionales		
No.	Nivel de Satisfacción	Observación

Nota. Realizado por los autores.

De esta forma se logra visualizar que la mayoría de los requerimientos planteados desde un principio y sus modificaciones durante las iteraciones, se llevaron a cabo con un nivel de satisfacción de más del 90%.

5.3.2 Accesibilidad a la Información

Actualmente el sistema se encuentra almacenado en los servidores de la universidad, se realizó un proceso de integración directamente con los administradores de servidores y se llevaron a cabo las pruebas necesarias para verificar el correcto funcionamiento.

Para ingresar al sistema, la persona encargada, debe acceder a través del enlace psvr.ufpso.edu.co y posteriormente diligenciar las credenciales en el formulario de ingreso. Una vez iniciada la sesión, el encargado podrá visualizar las imágenes capturadas por la cámara de videovigilancia, junto con las demás funcionalidades del sistema descritas en la sección de construcción del prototipo de reconocimiento facial.

La instalación de la cámara de videovigilancia se realizó en el laboratorio de redes y telecomunicaciones a través de una conexión directa a la intranet de la UFPSO, debido a que el edificio de ingenierías no contaba con la infraestructura necesaria para llevar a cabo la instalación de la cámara.

El sistema adquiere las imágenes de vídeo a través de la cámara de videovigilancia por medio de la dirección IP bajo el protocolo de transmisión en tiempo real (RTSP), el sistema captura cada fotograma de la transmisión y realiza un proceso (asíncrono) de análisis fotograma por fotograma para detectar rostros en la imagen, si los detecta, realiza el proceso (asíncrono) de identificación, por lo que al comparar el modelo de reconocimiento facial creado con los datos del modelo del clasificador, se determina si la persona es o no conocida.

Si la persona es conocida, el sistema realiza una consulta a la base de datos construida en MySQL, la consulta se realiza por medio de un identificador, el cual retorna los datos personales que se muestran en pantalla, estos datos son capturados por el sistema internamente y crea una máscara sobre los píxeles de la imagen original, con los datos de la persona identificada.

Los módulos de gestión del sistema, están intercomunicados entre sí con el servidor por medio de WEB Sockets, por lo que cualquier cambio realizado se refleja en tiempo real. Por otra parte, las bases de datos son independientes, pero están alojadas en el mismo servidor, por lo que a través de un endpoint suministrado por la base de datos principal de la universidad, se pueden almacenar los datos de integración necesarios para la base de datos del sistema de reconocimiento facial, teniendo una paridad de la data entre ambas bases de datos.

5.3.3 Escalabilidad

5.3.3.1 Infraestructura

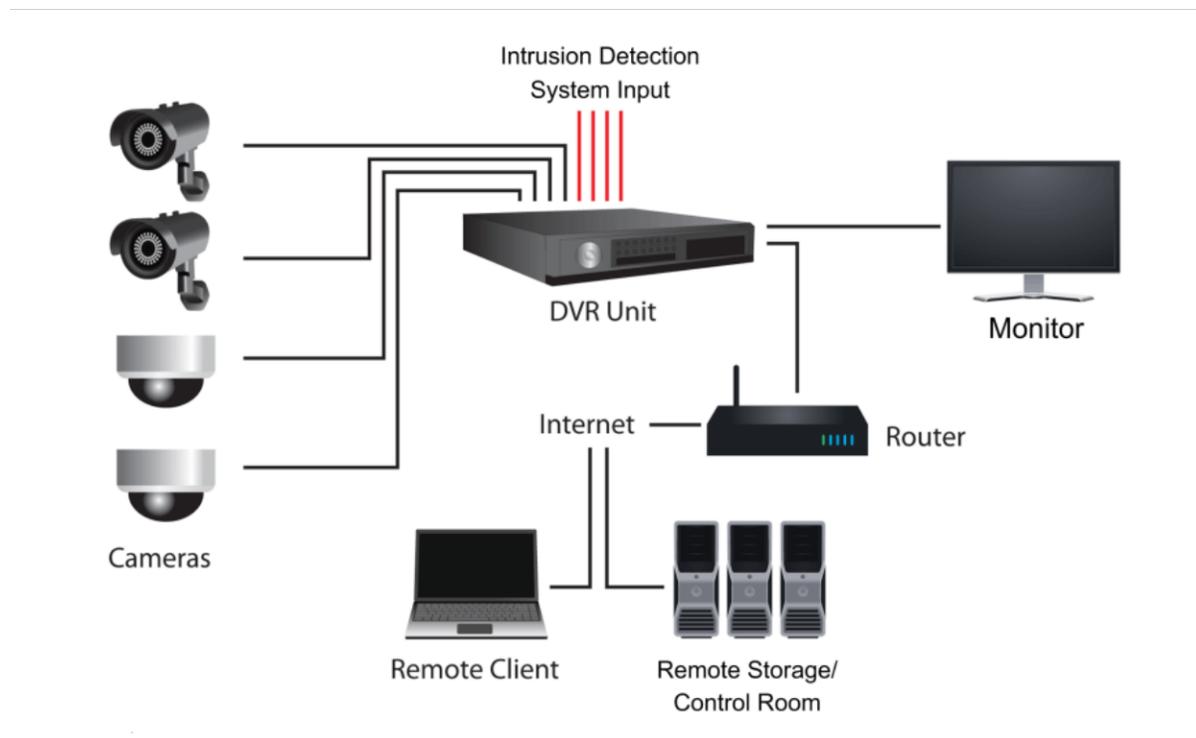
La infraestructura de escalabilidad estaría determinada por el conjunto de herramientas de hardware y software del circuito cerrado de televisión o CCTV.

Se debe seleccionar un conjunto de cámaras de acuerdo con las necesidades del entorno, es decir, si se necesita capturar la imagen de una zona de control al aire libre o una zona de control en interiores, una vez establecido el tipo de cámara, que pueden ser:

- Cámara Domo.
- Cámara Bullet.
- Cámara Anti Balística.
- Cámara Motorizada.

Estas cámaras deben contar con la tecnología IP, es decir, deben ser de tipo digital, al ser de tipo digital, deben estar conectadas a través de un cable UTP a un grabador NVR (Network Video Recorder), el grabador se encarga de interconectar todas las señales de vídeo de las cámaras, de acuerdo con la cantidad de cámaras, se estimaría el tipo de NVR necesario para la instalación. A su vez, el NVR se encarga de almacenar las imágenes de vídeo de las cámaras, mostrarlas a través de pantallas de vídeo y de enviar las imágenes a través de un puerto internet a cualquier tipo de aplicación que se integre.

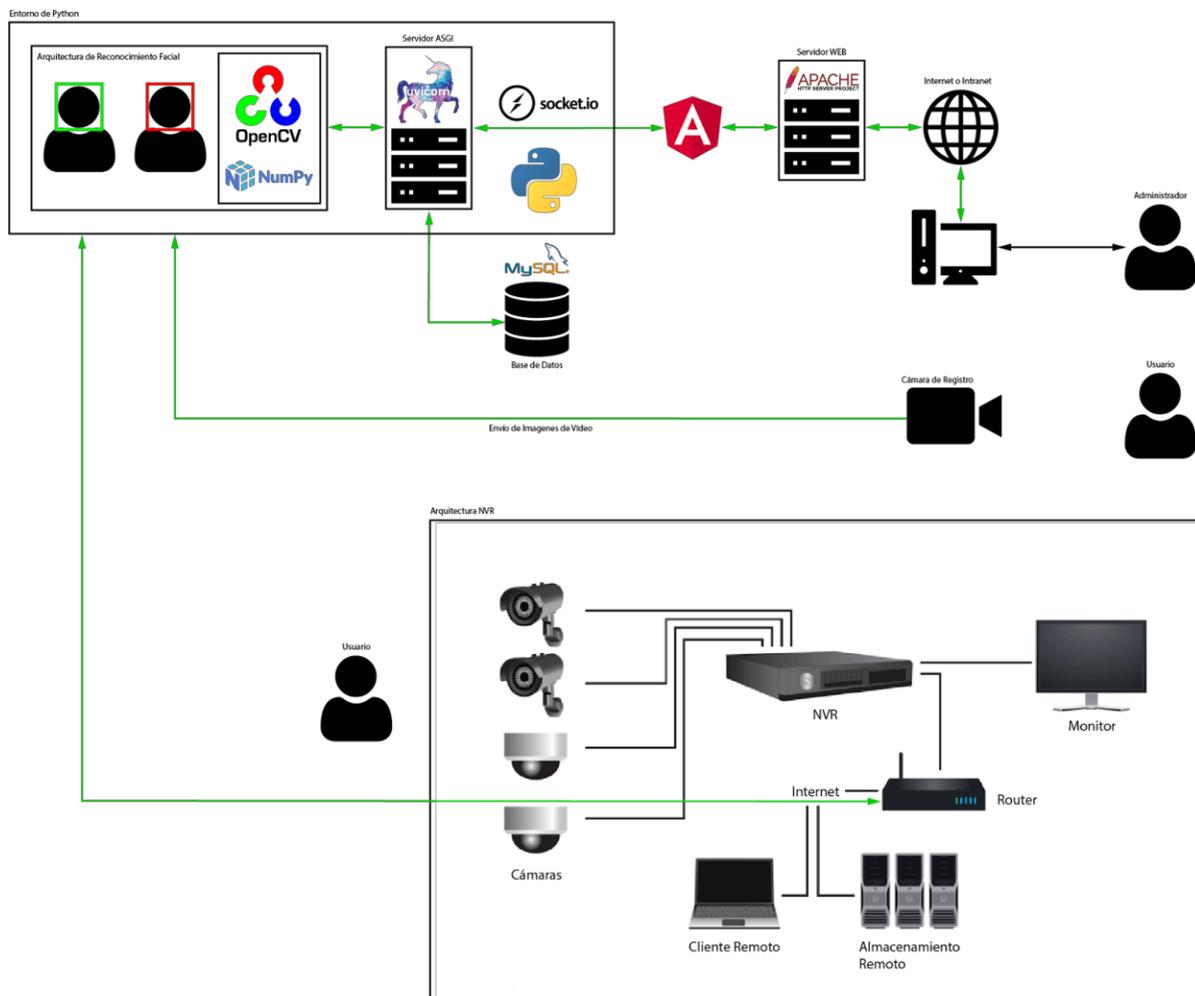
Básicamente un NVR funciona como un router, pero enfocado a la parte transmisión de imágenes de vídeo, por lo que un sistema de reconocimiento facial escudo, debería establecerse bajo una arquitectura de CCTV de la siguiente manera:

Figura 159*Sistema NVR*

Nota. Datos tomados de (Security360, 2022).

Si no se desean almacenar las imágenes capturadas por las cámaras de videovigilancia, no es necesaria la adecuación de un NVR dentro de la arquitectura, por lo que cámaras podrían ir conectadas directamente a un switch o a un router dentro de una arquitectura de red.

Figura 160*Arquitectura CCTV y el PSR*



Nota. Realizado por los autores.

5.3.3.2 Almacenamiento

La forma de reconocimiento depende mucho de la base de datos que se utilice y el tipo de clasificador, por lo que, si se utiliza una base de datos no discriminatoria sin cambios de luces y características de personas, sumado al tipo de técnica de reconocimiento, el reconocimiento será o no adecuado.

A pesar de que se desconoce el tipo de hardware utilizado por el servidor de la universidad, se puede realizar una estimación de la capacidad de almacenamiento y procesamiento requerido para tratar el volumen de información del sistema de reconocimiento facial.

Actualmente, la universidad en el segundo semestre del 2022, cuenta con 5.764 estudiantes, 872 administrativos y 270 docentes, un total de 6.906 personas que deben ser registradas en el sistema de reconocimiento facial.

Con base en la cantidad total de personas adscrita a la universidad y teniendo en cuenta que la documentación de OpenCV recomienda tener como mínimo 8 fotos para reconocer una persona, se puede realizar un cálculo de escalabilidad para la cantidad de almacenamiento necesaria configurada para el servidor, teniendo:

- Comunidad Universitaria Adscrita (x)
- Constante de Fotos Mínimas Por Persona Para el Reconocimiento Facial (y)
- Total, de Fotos Mínimas de Partida Para el Funcionamiento del Sistema (z)

$$x * y = z$$

$$(6.906)(8) = 55.248 \text{ Fotos}$$

Para el funcionamiento mínimo del sistema de reconocimiento facial desde el segundo semestre del 2022, se deberían almacenar un total de 55.248 fotos y para efectuar una

escalabilidad en semestres posteriores, se deberían analizar los registros de las personas almacenadas con las nuevas a registrar en el sistema.

Las personas que no se encuentran registradas en el sistema deben pasar por el proceso de adquisición de rasgos biométricos, sumándose al total de fotos previamente almacenadas en el conjunto de datos de la base de datos, por lo que se tendría que:

- Constante de Fotos Mínimas Por Persona Para el Reconocimiento Facial (y)
- Fotos Registradas en el Sistema de Reconocimiento Facial del Semestre Anterior (u)
- Comunidad Universitaria Nueva (No Registrada) Por Semestre (v)
- Total, de Fotos de Personas Registradas en el Sistema de Reconocimiento Facial (w)

$$u + vy = w$$

Por ejemplo, si para el siguiente semestre se requirieran almacenar 3000 personas nuevas en el Sistema de Reconocimiento Facial, se tendría:

$$55.248 + (3.000)(8) = 79.248 \text{ Fotos}$$

Teniendo en cuenta esto, se podría realizar un estudio de escalabilidad del sistema, con base la trazabilidad de personas registradas en los semestres anteriores, proyectando la cantidad aproximada de almacenamiento requerida para el servicio.

Además, se debe tener en cuenta que de acuerdo con el tipo de resolución de la foto de donde se extrae el rostro de la persona, puede existir una variabilidad en el tamaño de almacenamiento (peso) de la misma. Teniendo en cuenta esto, gracias a la configuración establecida en el sistema desarrollado, la foto extraída del rostro tiene un peso aproximado de 8KB a 12KB, por lo que, para un sistema de reconocimiento facial, con datos actuales, se requeriría mínimo para el registro un total de:

$$12KB)(55.248) = 662.976KB, \text{ es decir: } 662.976MB, \text{ es decir: } 0.66GB$$

Por otra parte, para el control de personas desconocidas, si el sistema reconoce que hay una persona desconocida, el sistema realiza una captura en el momento en que detecta a la persona y cada cierto tiempo, si la persona desconocida aún se encuentra en el lugar donde está la cámara de videovigilancia, se capturará una nueva foto de la persona, almacenándose en un buffer de desconocidos que serán identificados con el mínimo de fotos por persona por medio de un código de desconocido, por lo que se realiza un entrenamiento de desconocidos, haciendo que si la persona desconocida fue identificada con anterioridad no se almacene un volumen de información de fotos innecesarias, aligerando el almacenamiento en el servidor.

Teniendo en cuenta que una persona con mínimo 8 fotos ocupa un espacio de almacenamiento de 96KB, se puede establecer un porcentaje adicional a la cantidad de almacenamiento total, manejando un margen de fotos adicionales de desconocidos o falsos positivos.

El porcentaje de almacenamiento adicional dependerá en gran medida de la estimación que se realice con relación al total de fotos mínimas, por lo que un porcentaje de un 10% del total de fotos mínimas, sería idóneo.

Siguiendo el ejemplo y aplicando este porcentaje al total de personas, tendríamos que habría un margen de almacenamiento de:

$$6.906 * 10\% = 690 \text{ Personas}$$

Por lo que se tendrían:

$$(690)(8) = 5.520 \text{ Fotos}$$

Y el tamaño de almacenamiento adicional sería:

$$(12KB)(5.520) = 66.240KB \text{ es decir: } 66.24MB, \text{ es decir: } 0.066GB$$

En conclusión se puede definir un almacenamiento mínimo para que el sistema funcione, en relación con la cantidad de usuarios que se registren en el sistema de reconocimiento facial, por lo que es necesario contar con un sistema de gestión de almacenamiento automatizado que se encargue de asignar recursos de almacenamiento de acuerdo con el volumen de información necesaria a almacenar, en llegado caso que no se cuente con este sistema, la evaluación debería realizarla la persona encargada de administrar el sistema semestre a semestre.

5.3.3.3 Procesamiento de Vídeo y Procesamiento Computacional

Para el procesamiento de vídeo, es necesario contar una tarjeta gráfica especializada en cálculos matemáticos paralelos, por lo que se debería optar por una GPU orientada a este tipo de procesamiento, en el mercado existe una gran variedad de GPUs con estas características, se podría tener en consideración la GPU NVIDIA A100, especializada en el mundo de la inteligencia artificial.

La GPU NVIDIA A100 Tensor Core ofrece una aceleración sin precedentes en todas las escalas para impulsar los data centers elásticos de mayor rendimiento del mundo para inteligencia artificial, análisis de datos y HPC. Desarrollado por la arquitectura NVIDIA Ampere, la A100 es el motor de la plataforma de la data center NVIDIA. La A100 proporciona un rendimiento hasta 20 veces mayor que la generación anterior y se puede dividir en hasta siete instancias de GPU para ajustarse dinámicamente a las demandas cambiantes. Disponible en versiones de memoria de 40GB y 80GB, A100 80GB presenta el ancho de banda de memoria más rápido del mundo a más de 2 terabytes por segundo (TB/s) para ejecutar los modelos y conjuntos de datos más grandes.

Para el procesamiento computacional es necesario contar con un procesador que evite el cuello de botella, por lo que se requiere uno que sea compatible con la GPU, con base en la estación de trabajo planteada por NVIDIA, se requeriría de un CPU Single AMD 772, 64 cores.

5.3.4 Tecnologías Utilizadas y Procedimientos Requeridos

Las tecnologías utilizadas fueron explicadas durante el desarrollo del trabajo, las principales fueron:

- OpenCV
- Python
- MySQL
- WebSockets
- Angular

Pasos de Integración:**Tabla 46***Herramientas de Software*

#	Requerimiento	Descripción	URL	Comando de instalación
1	Instalar Python en su versión 3.9.6.	Python es un lenguaje de alto nivel de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código.	https://www.python.org/	
2	Instalar NodeJS 14.17.3.	Node.js® es un entorno de ejecución para JavaScript construido con V8, motor de JavaScript de Chrome.	https://nodejs.org/es/	
3	Instalar Angular.	Angular es un framework de diseño de aplicaciones y plataforma de desarrollo para crear aplicaciones de una sola página eficientes y sofisticadas	https://angular.io/	<code>npm install -g @angular/cli</code>
4	Instalar la librería “uvicorn”.	Uvicorn es una implementación de servidor web ASGI para Python.	https://www.uvicorn.org/	<code>pip install uvicorn</code>
5	Instalar la librería “uvicorn[standard]”.	Uvicorn Standard es una implementación de servidor web ASGI para Python con dependencias mínimas (Python puro).	https://www.uvicorn.org/	<code>pip install uvicorn[standard]</code>
6	Instalar la librería “imutils”.	Una serie de funciones convenientes para hacer que las funciones básicas de	https://pypi.org/project/imutils/	<code>pip install imutils</code>

#	Requerimiento	Descripción	URL	Comando de instalación
		procesamiento de imágenes, como traducción, rotación, cambio de tamaño, esqueletización, visualización de imágenes de Matplotlib, clasificación de contornos, detección de bordes y mucho más, sean más fáciles con OpenCV y Python 2.7 y Python 3.		
7	Instalar la librería “numpy”	NumPy es una biblioteca para el lenguaje de programación Python que da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas.	https://numpy.org/install/	<code>pip install numpy</code>
8	Instalar la librería “opencv-python”	Paquete contenedor para enlaces de python OpenCV.	https://pypi.org/project/opencv-python/	<code>pip install opencv-python</code>
9	Instalar la librería “bcrypt”	Hashing moderno de contraseñas para su software y sus servidores	https://pypi.org/project/bcrypt/	<code>pip install bcrypt</code>
10	Instalar la librería “pyjwt”	Implementación de JSON Web Token en Python	https://pypi.org/project/PyJWT/	<code>pip install pyjwt</code>
11	Instalar la librería “opencv-contrib-python”	Paquete contenedor para enlaces de python OpenCV.	https://pypi.org/project/opencv-contrib-python/	<code>pip install opencv-contrib-python</code>
12	Instalar la librería “python-socketio”	Servidor y cliente Socket.IO para Python	https://pypi.org/project/python-socketio/	<code>pip install python-socketio</code>

#	Requerimiento	Descripción	URL	Comando de instalación
13	Instalar la librería con el conector de MySQL.	Controlador MySQL escrito en Python	https://pypi.org/project/mysql-connector-python/	<code>python -m pip install mysql-connector-python</code>
14	Crear la base de datos “psvr” de acuerdo con el modelo relacional.	Base de datos del sistema de reconocimiento facial.		

Nota. Realizado por los autores.

Una vez instalado todo el software requerido, ya se podría ejecutar el servicio de Angular por medio del comando (`ng serve [project]`) o se podría sacar la producción del sistema con el comando (`ng build [project]`) y cargarla directamente en el almacenamiento del servidor de la universidad, a su vez, se podría ejecutar el servicio de UVICORN (`uvicorn --reload [project.app]`) y el servicio de MySQL, para el funcionamiento de la base de datos.

A su vez, se debe realizar el proceso de configuración en el código de las IPs de las cámaras, para que el sistema identifique de dónde está extrayendo el recurso de vídeo.

Capítulo 6. Conclusiones

Por medio del estudio del estado del arte se logró establecer un punto de partida que permitiera analizar los diferentes tipos de biometrías, los sistemas de reconocimiento de patrones y las tecnologías necesarias para el desarrollo del prototipo. Asimismo, se logró establecer una arquitectura de sistema de reconocimiento facial que definiera las etapas del mismo.

Gracias a la metodología de prototipos, se logró llevar a cabo un proceso de levantamiento de requisitos acompañado de un estudio de la literatura, que permitió en primera instancia, tener un panorama de desarrollo, de acuerdo con los sistemas de reconocimiento biométrico. Por otra parte, por medio de la implementación de WebSockets y las librerías de visión por computadora OpenCV, se logró llevar a cabo la experimentación con el tratamiento de imágenes y la transmisión de las mismas a través de un servicio ASGI y un cliente Angular.

Finalmente, durante los procesos de análisis y pruebas de cada iteración, se lograron determinar las posibles inconsistencias o refinamientos necesarios, que permitieran el desarrollo efectivo del prototipo, siendo así que por cada iteración se generara un resultado particular que llevara a la estructuración de la iteración consecutiva, proyectando el resultado final de un prototipo funcional.

Capítulo 7. Recomendaciones

Para lograr una efectividad de la ejecución del prototipo, es necesario utilizar hardware robusto que permita la implementación del sistema de forma eficiente, debido a que la manipulación y transmisión de las imágenes de vídeo requiere de un nivel alto de procesamiento.

Es importante tener en cuenta que el sistema puede presentar inconsistencias debido a que es un prototipo (beta) de sistema de seguridad que puede o no ser implementado en la universidad. Es importante destacar que puede utilizarse como un medio para la experimentación de los estudiantes de la universidad.

Se recomienda tener en cuenta la sección de Escalabilidad, donde se explica a detalle el procedimiento necesario para implementar el sistema de reconocimiento facial dentro de un circuito cerrado de televisión (CCTV), distribuido en la UFPSO.

Para fomentar el desarrollo del prototipo, es recomendable lanzar el sistema al estudiantado o ente desarrollador por medio de un controlador de versiones que permita llevar las pautas del desarrollo del proyecto, por esta razón, durante el proceso de desarrollo del prototipo se utilizó GitHub, ya que permitía efectuar la auditoría de cada una de las versiones del prototipo, controlando el desarrollo y modificación del código de cada iteración.

Capítulo 8. Referencias

- Alcalá Nalvaiz, J. T., & Campo León, E. (Septiembre de 2022). *Universidad de Zaragoza*.
Obtenido de Introducción a las máquinas de vector soporte (SVM) en aprendizaje supervisado: <https://zaguan.unizar.es/record/59156/files/TAZ-TFG-2016-2057.pdf>
- A. Melnikov, I. F. (12 de 2011). Obtenido de hjp: <https://www.hjp.at/doc/rfc/rfc6455.html>
- Abdi, H., & Williams, L. J. (2010). Principal component analysis.
- Abuzneid, M. A., & Mahmood, A. (2018). Enhanced Human Face Recognition Using LBPH Descriptor, Multi-KNN, and Back-Propagation Neural Network. *IEEE Access*, 20641 - 20651.
- Alegre Gutiérrez, E., Pajares Martinsanz, G., & de la Escalera Hueso, A. (2016). *Conceptos y métodos en Visión por Computador*. España: Grupo de Visión del Comité Español de Automática (CEA).
- Alegre, E., Pajares, G., & De la Escalera, A. (Junio de 2016). *Grupo de visión del comité Español de Automatica*. Obtenido de Conceptos y metodos en visión por computador: <https://intranet.ceautomatica.es/sites/default/files/upload/8/files/ConceptosyMetodosenVxC.pdf>
- Alfonso, M., Cazorla, M., Colomina, O., Escolano, F., & Lozano, M. (2003). Inteligencia artificial: modelos, técnicas y áreas de aplicación. En M. Alfonso, M. Cazorla, O. Colomina, F. Escolano, & M. Lozano, *Inteligencia artificial: modelos, técnicas y áreas de aplicación* (pág. 370). Alicante: Paraninfo.
- Alonso, F., Martínez, L., & Segovia, F. (2005). *INTRODUCCION A LA INGENIERIA DEL SOFTWARE: MODELO DE DESARROLLO DE PROGRAMAS*. Delta Publicaciones.

- Amat, J. (Septiembre de 2018). *Ciencia de Datos*. Obtenido de https://www.cienciadedatos.net/documentos/50_algoritmo_perceptron
- Amat, R. (2020). Análisis discriminante lineal (LDA) y análisis discriminante cuadrático (QDA). *Attribution 4.0 International*.
- Amazon. (2022). *Amazon*. Obtenido de ¿Qué es el boosting?: <https://aws.amazon.com/es/what-is/boosting/>
- AMD. (2021). Obtenido de AMD: <https://www.amd.com/es/products/cpu/amd-ryzen-9-3900x>
- Amorim, W. P., & Pistori, H. (2007). Análise Discriminante de Fisher Aplicada à Detecção de Defeitos em Couro Bovino.
- Anguas Pérez, J. M. (2020). *Análisis y diseño de dispositivos de radiocomunicación mediante*. Sevilla. Obtenido de <http://bibing.us.es/proyectos/abreproy/11084/fichero/Memoria+por+cap%C3%ADtulos+%252FCap%C3%ADtulo+4.pdf>
- Basogain Olabe, X. (s.f.). *OpenCourseWare (OCW)*. Recuperado el Agosto de 2022, de Redes neuronales artificiales y sus aplicaciones.
- Caldera, J., & Zapico, F. (2009). Identificación facial biométrica. *MiBiblioteca*.
- Cardona Alzate, N. I., Ospina Arango, J. D., & Correa Morales, J. C. (Septiembre de 2019). *Universidad Nacional de Colombia*. Obtenido de Predicción y selección de variables con bosques aleatorios en presencia de variables correlacionadas: <https://repositorio.unal.edu.co/bitstream/handle/unal/75561/8063120.2019.pdf?isAllowed=y&sequence=1>
- Carranza Astrada, P. (2015). *Universidad Nacional de Cordoba*. Obtenido de Reconocimiento de caracteres en imágenes no estructuradas: <https://rdu.unc.edu.ar/handle/11086/2818>

Castaño Saavedra, D., & Alonso Sierra, J. D. (2019). *Universidad Católica de Colombia*.

Obtenido de SISTEMA DE RECONOCIMIENTO FACIAL PARA CONTROL DE ACCESO A VIVIENDAS:

<https://repository.ucatolica.edu.co/bitstream/10983/24032/1/Final%20Trabajo%20de%20grado.pdf>

CGN. (2017). *cg*n. Obtenido de

https://www.cgn.gub.uy/innovaportal/file/83018/1/material_concurso_r14_cgn_2017.pdf

CORSAIR. (2021). *Corsair Gaming, Inc*. Obtenido de Corsair Gaming, Inc Web Site:

<https://www.corsair.com/lm/es/Categor%C3%ADas/Productos/Unidades-de-fuentes-de-alimentaci%C3%B3n/Unidades-de-fuentes-de-alimentaci%C3%B3n-avanzadas/RMx-Series/p/CP-9020180-NA>

Cuevas Alfaro, E. A. (Junio de 2010). *PONTIFICIA UNIVERSIDAD CATÓLICA DE*

VALPARAÍSO. Obtenido de “MÁQUINAS DE SOPORTE VECTORIAL CON ALGORITMOS BASADOS EN POBLACIONES PARA EL PRONÓSTICO DEL PRECIO DE ACCIONES LAN CHILE: http://opac.pucv.cl/pucv_txt/Txt-4500/UCG4828_01.pdf

Dev47Apps. (1 de Octubre de 2021). Obtenido de Dev47Apps - DroidCam:

<http://www.dev47apps.com/>

EcuRed. (21 de Enero de 2013). Obtenido de EcuRed:

https://www.ecured.cu/Modelo_de_prototipos

EcuRed. (2021). Obtenido de https://www.ecured.cu/Modelo_de_prototipos

Esparza Franco, C., Tarazona Ospina, C., Sanabria Cuevas, E., & Velazco Capacho, D. (2015).

Revista Colombiana de Tecnologías de Avanzada. Obtenido de RECONOCIMIENTO

FACIAL BASADO EN EIGENFACES, LBHP Y FISHERFACES EN LA
BEAGLEBOARD-xM:

https://www.researchgate.net/publication/319958509_RECONOCIMIENTO_FACIAL_BASADO_EN_EIGENFACES_LBHP_Y_FISHERFACES_EN_LA_BEAGLEBOARD-xM

Fernandez Casal, R. (Agosto de 2022). *R Machinery*. Obtenido de Bosques aleatorios:

https://rubenfcasal.github.io/aprendizaje_estadistico/bosques-aleatorios.html

García Monsálvez, J. C. (2017). Python como primer lenguaje de programación textual en la Enseñanza Secundaria. *Education in the Knowledge Society*, 147 - 162.

García Mora, T., & Mas, J. (2008). *Investigaciones Geográficas, Boletín del Instituto de Geografía, UNAM*. Obtenido de Comparación de metodologías para el mapeo de la cobertura y uso del suelo en el sureste de México:

https://www.researchgate.net/publication/38104495_Comparacion_de_metodologias_para_el_mapeo_de_la_cobertura_y_uso_del_suelo_en_el_sureste_de_Mexico

García Santillán, I. D., & Chacua Criollo, B. E. (2019). *Universidad Técnica del Norte*. Obtenido de Diseño de un sistema prototipo de reconocimiento facial para la identificación de personas en la Facultad de Ingeniería en Ciencias Aplicadas (FICA) de la Universidad Técnica del Norte utilizando técnicas de Inteligencia Artificial:

<http://repositorio.utn.edu.ec/handle/123456789/9572>

Gimeo Hernandez, R., & Morros Rubio, J. (Mayo de 2010). *Universidad Politecnica de Catalunya*. Obtenido de ESTUDIO DE TÉCNICAS DE RECONOCIMIENTO FACIAL.

GitHub. (2020). Obtenido de Octoverse GitHub: <https://octoverse.github.com/>

GitHub, Inc. (2021). Obtenido de GitHub Web Site: <https://octoverse.github.com/>

- Goldberg, D. E., & Holland, J. H. (1988). Genetic Algorithms and Machine Learning. *Machine Learning* 3, 95-99.
- Gómez Gil, M., & Sucar Succar, E. (2019). *Academia Mexicana de Computación*. doi:978-607-97357-7-7
- Google. (22 de 11 de 2021). Obtenido de Angular: <https://angular.io/docs>
- Gutiérrez González, L. (2000). *Escuela de Ingeniería, Universidad de las Américas Puebla*.
Obtenido de Marco teorico:
http://catarina.udlap.mx/u_dl_a/tales/documentos/msp/gutierrez_g_l/capitulo2.pdf
- Hangzhou Hikvision Digital Technology Co., Ltd. (2021). Obtenido de HikVision Web Site:
<https://www.hikvision.com/es-la/products/IP-Products/Network-Cameras/Pro-Series-EasyIP-/DS-2CD2183G0-I-S-/?q=ds-2cd2183g0-i-s-&position=23>
- Hernández, R. (2014). *Metodología de la Investigación*. México : INTERAMERICANA EDITORES, S.A. DE C.V.
- Hmong. (2022). Obtenido de Clasificador lineal: https://hmong.es/wiki/Linear_classifier
- INRED. (2007). *Instituto de Recursos Naturales y Ecodesarrollo*. Obtenido de Sistemas de Información Geográfica y Ordenación Territorial:
https://bibliotecavirtualaserena.files.wordpress.com/2017/11/2_2_clasificacic3b3n.pdf
- Ipanaqué, J. (2020). *ANÁLISIS COMPARATIVO DE TÉCNICAS DE*. Pimentel.
- Kivy. (2021). *Kivy*. Obtenido de <https://kivy.org/#home>
- Massiris, M., Delrieux, C., & Fernández Muñoz, J. Á. (2018). Detección de equipos de protección personal mediante red neuronal convolucional YOLO. En *Actas de las XXXIX Jornadas de Automática, Badajoz, 5-7 de Septiembre de 2018* (págs. 1022-1029).
Badajoz.

- Mauricio, S. (2020). *Reconocimiento de sujetos y caracterización de sus emociones a partir de sus expresiones*. Medellín.
- Neural Networks Framework*. (2020). Obtenido de <http://www.redes-neuronales.com.es/tutorial-redes-neuronales/el-perceptron-simple.htm>
- Nicolas, L., & Toro, J. (2012). *TECNICAS DE BIOMETRIA BASADAS EN PATRONES FACIALES DEL SER HUMANO*. Pereira.
- Noguera, C. (2012). *Autenticación por Reconocimiento Facial Para Aplicaciones WEB, Utilizando Software Libre*. Bucaramanga.
- NVIDIA. (2021). Obtenido de NVIDIA: <https://www.nvidia.com/es-la/drivers/what-is-gpu-computing>
- NVIDIA. (2021). *NVIDIA Corporation*. Obtenido de NVIDIA Web Site: <https://www.nvidia.com/es-la/geforce/graphics-cards/rtx-2060-super/>
- OpenCV. (2021). *OpenCV*. Obtenido de <https://opencv.org/about/>
- OpenCV. (2022). *OpenCV*. Obtenido de Face Recognition with OpenCV: https://docs.opencv.org/4.x/da/d60/tutorial_face_main.html#tutorial_face_lbph
- Ortega García, J., Alonso Fernández, F., & Coomonte Belmonte, R. (2008). *Biometría y seguridad*. Madrid: Fundación Rogelio Segovia para el Desarrollo de las Telecomunicaciones.
- Ospina Mateus, E. (2008). *Universidad Tecnología de Bolívar*. Obtenido de Hidden Markov Models (HMM's) y aplicaciones: <https://biblioteca.utb.edu.co/notas/tesis/0049845.pdf>
- Palomera, D., & Figueroa, A. (Diciembre de 2015). *UNIVERSIDAD ANDRÉS BELLO*. Obtenido de Aprendizaje semi-supervisado para la detección de respuestas informativas en preguntas procedurales:

- https://www.researchgate.net/publication/310374887_Aprendizaje_semi-supervisado_para_la_deteccion_de_respuestas_informativas_en_preguntas_procedurales
- Paz Arias, H., & Jimenez, D. (Agosto de 2018). *Revista Tecnológica ESPOL*. Obtenido de Desarrollo de un sistema inteligente para la clasificación de documentos ya digitalizados aplicando redes neuronales supervisadas:
- https://www.researchgate.net/publication/281380920_Desarrollo_de_un_sistema_inteligente_para_la_clasificacion_de_documentos_ya_digitalizados_aplicando_redes_neuronales_supervisadas
- Pedraza Pico, B. O., Rondón, P., & Arguello, H. (2011). Sistema de reconocimiento facial basado en imágenes con color. *Revista UIS Ingenierías*, 113-122.
- Pérez, Á. (2017). *Verificador facial en dispositivos móviles, y estrategias de fusión de biometría facial y de voz*. Madrid.
- Plata, R. (2020). MatLab & Redes Neuronales. *Revistas Bolivianas*.
- Portilla Jiménez, J. J. (2018). Análisis y diseño de un sistema de reconocimiento facial aplicando machine learning para detectar e identificar intrusos. (*Tesis*). Universidad de Guayaquil.
- Preparadores de oposiciones. (2022). *Preparadores de oposiciones*. Obtenido de La voz humana: Características, clasificación y fisiología: <https://www.preparadores.eu/la-voz-humana>
- Razo Gil, L. J. (2009). Sistema para el reconocimiento del alfabeto dactilológico. (*Tesis*). Instituto politécnico nacional, Ciudad de México.
- Riverón, C., & Marroig, L. (2011). *Universidad de la republica de Uruguay*. Obtenido de RECONOCIMIENTO DE CARAS Y DESEMPEÑO DE CLASIFICADORES: https://eva.fing.edu.uy/file.php/514/ARCHIVO/2011/TrabajosFinales2011/informe_final_riveron_marriog.pdf

- Rosa Pérez, E., Liberatore, F., & Cabezón Manchado, M. (2018). *Universidad Complutense de Madrid*. Obtenido de Predicción demanda eléctrica española. Implementación de redes neuronales recurrentes en Python: <https://eprints.ucm.es/id/eprint/49444/1/2018-MIGUEL%20CABEZON%20Memoria.pdf>
- Ruiz, C. A., Basualdo, M. S., & Matich, D. J. (Marzo de 2001). *Universidad Tecnológica Nacional – Facultad Regional Rosario*. Obtenido de Redes Neuronales: Conceptos Básicos y : https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientadora1/monograiias/matich-redesneuronales.pdf
- Sailema, L., & Chango, G. (Diciembre de 2012). *TIC'S y Sociedad*. Obtenido de Desarrollo de un Sistema de Información para Monitoreo: <file:///E:/Downloads/Dialnet-DesarrolloDeUnSistemaDeInformacionParaMonitoreoYSe-6163753.pdf>
- Salas, J. (2018). *Instituto Politécnico Nacional*. Obtenido de Reconocimiento de Patrones: <https://www.cicataqro.ipn.mx/cq/qro/Paginas/index.php/component/jdownloads/send/3-profesores/prnotes4course01a.pdf>
- Salud Ocular. (2022). *Salud Ocular*. Obtenido de Claves para entender la visión III: el cerebro: <https://saludocular.com/tag/corteza-visual/#:~:text=La%20corteza%20visual%20es%20la,que%20ocurre%20dentro%20del%20cerebro>
- Santa Catarina, R., Marmolejo Cortes, D. F., Fiorio Vettorazzi, J. C., & Gonzaga Pereira, M. (Noviembre de 2018). *Australian Journal of Crop Science*. Obtenido de Image-based phenotyping of morpho-agronomic traits in papaya fruits (Carica papaya L. THB var.):

https://www.researchgate.net/figure/Digital-image-processing-with-ImageJ-software-a-Original-RGB-color-image-of-papaya_fig1_329488895

- Sanz, C. (08 de 2022). *Repositorio de la Universidad Nacional de La Plata*. Obtenido de Clasificación y reconocimiento de patrones. Estado del arte en “remote sensing”: http://sedici.unlp.edu.ar/bitstream/handle/10915/2213/4_-_Clasificaci%C3%B3n_y_reconocimiento_de_patrones._Estado_del_arte_en_%E2%80%9Cremote_sensing%E2%80%9D.pdf?sequence=7&isAllowed=y
- Security360. (2022). *Security360*. Recuperado el 18 de Octubre de 2022, de what are the 7 components of CCTV camera?: <https://security360.pk/components-of-cctv-camera/>
- Serrano, H. (2019). Obtenido de Universidad Don Bosco: https://www.udb.edu.sv/udb_files/recursos_guias/informatica-ingenieria/sistemas-expertos-e-inteligencia-artificial/2019/i/guia-1.pdf
- Serrano, H. (2019). Obtenido de Universidad Don Bosco: https://www.udb.edu.sv/udb_files/recursos_guias/informatica-ingenieria/sistemas-expertos-e-inteligencia-artificial/2019/i/guia-1.pdf
- Serratosa, F. (2008). La biometría para la identificación de las personas. *Universitat Oberta de Catalunya*, 8-20.
- The QT Company. (2020). *qt.io*. Obtenido de <https://www.qt.io/qt-for-python#:~:text=Qt%20for%20Python%20is%20the,PySide2%20module%20is%20available%20now!>
- Tipping, M. E., & M. Bishop, C. (2002). Probabilistic Principal Component Analysis. *Royal Statistical Society*.
- Tkiter. (2021). *Tkinter*. Obtenido de <https://docs.python.org/3/library/tkinter.html>

Udemy. (2022). *Udemy*. Obtenido de Python de cero hasta reconocimiento facial:

<https://www.udemy.com/join/login-popup/?next=/course/python-de-cero-hasta-reconocimiento-facial/learn/lecture/26084548#overview>

Universidad Politecnica de Valencia. (2017). Obtenido de ALGORITMO DE

RECONOCIMIENTO DE PATRONES BASADO EN CODIFICACION FISIOLÓGICA EN CEREBRO DE PRIMATES:

<https://riunet.upv.es/bitstream/handle/10251/92671/CASTEL%20-%20Algoritmo%20de%20reconocimiento%20de%20patrones%20basado%20en%20codificaci%C3%B3n%20fisiol%C3%B3gica%20en%20cerebro%20de....pdf?sequence=1>

Uvicorn. (2021). Obtenido de Uvicorn: <https://www.uvicorn.org/>

van Rossum, G. (2020). *Python Tutorial*.

Vázquez López , M. Á., & Cuevas de la Rosa , F. J. (Sistema de Reconocimiento Facial

Mediante de Marzo de 2017). *Centro de investigaciones en optica*. Obtenido de

<https://cio.repositorioinstitucional.mx/jspui/bitstream/1002/436/1/15950.pdf>

Vazquez, A. M. (2022). *Universidad Nacional Autonoma de Mèxico*. Obtenido de

INTRODUCCION AL RECONOCIMIENTO DE PATRONES: http://profesores.fi-b.unam.mx/ana/APUNTES_RP/capitulo1.pdf

Vega Huerta, H., Cortez Vásquez, A., Huayna, A. M., Alarcón Loayza, L., & Romero Naupari,

P. (Diciembre de 2009). *Revistas de investigación Universidad Nacional Mayor de San*

Marcos. Obtenido de Reconocimiento de patrones mediante redes neuronales artificiales:

<https://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/3301/2750>

Viola, P., & Jones, M. (2001). Robust Real-Time Face Detection. *International Journal of*

Computer Vision, 137-154.

Wikiwand. (Agosto de 2022). *Wikiwand*. Obtenido de Reconocimiento de patrones:

https://www.wikiwand.com/es/Reconocimiento_de_patrones

Zakaria, J. (5 de Diciembre de 2020). Obtenido de Builtin: [https://builtin.com/data-science/step-](https://builtin.com/data-science/step-step-explanation-principal-component-analysis)

[step-explanation-principal-component-analysis](https://builtin.com/data-science/step-step-explanation-principal-component-analysis)