

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



TESIS DE GRADO

BIOMETRÍA FACIAL BASADA EN REDES NEURONALES APLICANDO

EL ALGORITMO DE VIOLA – JONES

Para optar al Título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: GESTIÓN Y PRODUCCIÓN

Postulante: Blanca Stefani Machaca Ticona

Tutor Metodológico: Ing. Maricel Yarari Mamani

Tutor Especialista: Ing. Wilfredo Choque Cordero

Tutor Revisor: Ing. Ramiro Kantuta Limachi

EL ALTO – BOLIVIA

2020

Dedicatoria

A Dios por darme vida, salud, sabiduría y haberme guiado en mi camino.

A mis padres Walter Machaca (+) e Marcelina Ticona que sin ellos no hubiera logrado una meta más en mi vida profesional, gracias por todo el cariño y apoyo incondicional a lo largo de toda mi vida

A mi hermano Juan Pablo Pinto Ticona, mi hermana Eva Pinto Ticona por la compañía y el apoyo que siempre me brindaron.

A mis Sobrinas Brenda Pinto y Griselda Pinto por tanto amor y cariño que me brindaron de mis pequeñas Mellis.

A mis tutores, amigos y a todos aquellos que me apoyaron en el desarrollo de este trabajo.

Agradecimientos

Agradecer primero a Dios por todas las bendiciones recibidas en mi vida, por darme salud y la fuerza para poder culminar mi tesis de grado, por darme la oportunidad de seguir adelante con mi meta y por poner siempre a las personas correctas en el transcurrir de mi vida.

A mis estimados tutores mis más sinceros agradecimientos, agradecer a mi Tutor Especialista Ing. Wilfredo Choque, Tutor Metodológico Ing. Maricel Yarari Mamani y Tutor Revisor Ing. Ramiro Kantuta Limachi por compartir sus conocimientos, brindarme su orientación y sugerencias con paciencia y comprensión durante el desarrollo de la presente tesis.

Y finalmente a mis amigos y compañeros, por los momentos compartidos en aulas a lo largo de estos años y de esta manera lograr realizar nuestras metas.

INDICE

1. MARCO PRELIMINAR.....	1
1.1 Introducción	1
1.2 Planteamiento del Problema.....	2
1.2.1 Problema principal.....	2
1.2.2 Problemas secundarios	2
1.3 Objetivos.....	3
1.3.1 Objetivo general	3
1.3.2 Objetivo específico	3
1.4 Hipótesis	3
1.4.1 Identificación de variables	4
1.4.2 Operacionalización de variables.....	4
1.5 Justificación	5
1.5.1 Justificación científica.....	5
1.5.2 Justificación técnica.....	5
1.5.3 Justificación económica.....	5
1.5.4 Justificación social.....	6
1.6 Metodología	6
1.6.1 Método científico	6
1.6.2 Método de ingeniería.....	7
1.6.2.1 Metodología mixta	7
1.7 Herramientas	8
1.7.1 MatLab R2018b.....	8
1.7.2 Cámara web	9
1.7.3 XAMP	9

1.8	Límites y Alcances	9
1.8.1	Limites	9
1.8.2	Alcances	10
1.9	Aportes	10
2.	MARCO TEÓRICO	12
2.1	Ingeniería de Requerimientos	12
2.1.1	Requerimientos funcionales	12
2.1.2	Requerimientos no funcionales	13
2.1.3	Requisitos de Software.....	13
2.1.4	Requisitos de Hardware	13
2.2	Teoría Sobre RNA – Base Radial	14
2.2.1	Redes neuronales artificiales (RNA).....	14
2.2.1.1	Definiciones básicas de RNA	14
2.2.2	Elementos básicos de procesado.....	15
2.2.2.1	Sistema de patrones entrada – salida	15
2.2.2.2	Conexiones y pesos sinápticos	16
2.2.2.3	Funciones de activación	16
2.2.2.4	Elemento de procesado: neurona.....	18
2.2.2.5	Asociación de neuronas	18
2.2.3	Arquitecturas de RNA.....	20
2.2.3.1	Clasificación según número de capas	21
2.2.3.2	Clasificación según tipo de conectividad	21
2.2.4	Entrenamiento de RNA.....	23
2.2.4.1	Entrenamiento no supervisado	24
2.2.4.2	Entrenamiento supervisado.....	25

2.2.5	Redes neuronales artificiales de base radial	26
2.2.5.1	Arquitectura de las redes de base radial	28
2.2.5.2	Activación de las neuronas de la red de base radial	29
2.2.5.3	Aprendizaje de las redes de base radial.....	32
2.2.5.4	Método de aprendizaje híbrido	32
2.2.6	Redes de base radial frente a perceptrón multicapa	35
2.3	Teoría Sobre el Reconocimiento de Imágenes	37
2.3.1	Conceptos básicos de imágenes.....	37
2.3.2	Color	38
2.3.3	Teoría de procesamiento de imágenes	38
2.3.3.1	Conceptos generales.....	38
2.3.3.2	Vecindad entre píxeles.....	40
2.3.3.3	Conectividad.....	42
2.3.3.4	Relaciones y equivalencias.....	44
2.3.3.5	Filtrado convolucional	44
2.3.3.6	Segmentación.....	46
2.3.3.7	Thresholding.....	47
2.3.3.8	Crecimiento de regiones	47
2.3.3.9	Dilatar	49
2.3.3.10	Cerrar	49
2.3.4	Reconocimiento de patrones de una imagen	50
2.3.4.1	Etapas para el reconocimiento de patrones de una imagen.....	51
2.4	Teoría de Viola – Jones	54
2.4.1	Características.....	55
2.4.1.1	Imagen integral.....	55

2.4.1.2	Cálculo de imagen integral	56
2.4.1.3	Características Haar	56
2.5	Teoría de la Biometría	58
2.5.1	Biometría	59
2.5.2	Técnicas de autenticación	59
2.5.2.1	Combinando técnicas	60
2.5.3	Usos de Biometría	61
2.5.4	Funcionamiento básico de dispositivos biométricos	62
2.6	Teoría Sobre el Desarrollo del Prototipo	64
2.7	Métricas de calidad de software.....	65
2.8	Modelo de Costo Cocomo	66
2.8.1	Método de estimación de costo COCOMO II.....	66
2.8.1.1	Orgánico	66
2.8.1.2	Semi - acoplado.....	66
2.8.1.3	Empotrado	66
2.8.1.4	Modelo básico	67
2.8.1.5	Modelo intermedio	67
2.8.1.6	Modelo avanzado	67
2.8.1.7	Descripción de Valores Atributos de software	68
2.8.1.8	Atributos de hardware	68
2.8.1.9	Atributos del personal	69
2.8.1.10	Atributos del proyecto	69
2.9	Seguridad de la información	69
2.9.1	Estándar ISO/IEC 27000.	69
3.	MARCO APLICATIVO	70

3.1	Diagnóstico Actual de Ingeniería Autotrónica	70
3.2	Descripción de Ingeniería Autotrónica	70
3.2.1.1	Descripción del proceso de control del personal	71
3.3	Diagnóstico de la Topología y Estructura de RNA.	73
3.3.1	Tratamiento de las imágenes	74
3.3.1.1	Tamaño de imagen.....	76
3.3.1.2	Adecuación de imagen	77
3.3.2	Análisis de patrones faciales	79
3.3.2.1	Localización de los puntos característicos.....	79
3.3.2.2	Posición de los ojos, nariz y boca.....	81
3.3.2.3	Cálculo de distancia	82
3.3.2.4	Matriz característica de usuario.....	82
3.3.3	Descripción de las entradas	84
3.3.3.1	Entrenamiento de Red y resultados	85
3.4	Análisis y Diseño del Prototipo.....	88
3.4.1	Análisis	89
3.4.2	Casos de uso.....	89
3.4.3	Diagrama de secuencias	110
3.4.4	Diseño	111
3.4.5	Diagrama de interfaz	111
3.5	Validación del Sistema de Reconocimiento Facial.....	112
3.5.1	Matriz de confusión	112
3.6	Demostración del Prototipo.....	112
3.6.1	Implementación	112
4.	MÉTRICAS DE CALIDAD Y ESTIMACIÓN DE COSTOS	117

4.1	Métrica de Calidad del Software ISO/IEC 9126	117
4.1.1	Funcionalidad	117
4.1.2	Fiabilidad	117
4.1.3	Usabilidad.....	118
4.1.4	Mantenibilidad	118
4.1.5	Eficiencia	119
4.1.6	Portabilidad.....	119
4.1.7	Análisis de resultados.....	120
4.2	Estimación de Costos de Software	121
4.2.1	Teoría de costos.....	121
5.	PRUEBAS DE HIPÓTESIS.....	128
5.1	Prueba de Hipótesis T	128
5.2	Prueba de hipótesis F	129
6.	CONCLUSIONES Y RECOMENDACIONES.....	131
6.1	Conclusiones	131
6.2	Recomendaciones	131
6.3	Bibliografía.....	132
6.4	Webgrafía	134
	ANEXOS	135

ÍNDICE DE FIGURAS

Figura 1.1 Proceso de Investigación	6
Figura 2.1 Sigmoidea.	17
Figura 2.2 Estructura y funcionamiento de una neurona artificial.....	18
Figura 2.3 Asociación de neuronas por capas.	19
Figura 2.4 Esquema de clasificación de una RNA.	20
Figura 2.5 Diagrama de flujo de entrenamiento no supervisado.	24
Figura 2.6 Diagrama de flujo de entrenamiento supervisado.	25
Figura 2.7 Arquitectura de la red de neuronas de base radial.....	29
Figura 2.8 Coordenadas de una matriz.	39
Figura 2.9 Definiciones básicas.....	40
Figura 2.10 Píxeles adyacentes.	41
Figura 2.11 Vecindades, (a) vecindad de 4, (b) vecindad de 8.	42
Figura 2.12 Conectividad.	43
Figura 2.13 Matriz 3x3 pasa-altas.	45
Figura 2.14 Matriz 3x3 pasa-bajas	45
Figura 2.15 Operadores gradiente más comunes.	46
Figura 2.16 (a) Imagen ruidosa; (b) resultado de la detección de contornos.....	47
Figura 2.17 Geometría de crecimiento de regiones.....	48
Figura 2.18 Ejemplo de imagen dilatada.	49
Figura 2.19 Ejemplo de cerrar una imagen.	50
Figura 2.20 Patrones de reconocimiento.....	52
Figura 2.21 Diagrama de bloques.	55

Figura 2.22 Imagen integral.....	55
Figura 2.23 Cálculo de imagen integral.....	56
Figura 2.24 Filtros Haar rotados, trasladados y con cambios de escala.	57
Figura 2.25 Representación de filtros Haar.....	58
Figura 2.26 Cascada de clasificadores.	58
Figura 2.27 Sistemas biométricos.	62
Figura 2.28 Características de la norma ISO/IEC 9126.	65
Figura 3.1 Detalle de estudiantes.....	71
Figura 3.2 Parte diario de las actividades Ingeniería Autotronica.....	72
Figura 3.3 Estructura de la RNA.....	73
Figura 3.4 Tratamiento de imágenes.....	74
Figura 3.5 Tamaño y resolución de una imagen.	76
Figura 3.6 Componentes de una imagen.	77
Figura 3.7 Análisis facial (quintos).....	79
Figura 3.8 Detectando rasgos faciales.	79
Figura 3.9 Detección de características de un rostro.	81
Figura 3.10 Puntos encontrados en un rostro.	82
Figura 3.11 Matriz de características	83
Figura 3.12 Graficando newrb (MatLab).....	87
Figura 3.13 Resultado del entrenamiento a la RBF.....	88
Figura 3.14 Casos de uso	89
Figura 3.15 Inicio de sesion.....	112
Figura 3.16 Pantalla principal.	113

Figura 3.17 Pantalla principal.....	113
Figura 3.18 Registro de docente nuevo.	114
Figura 3.19 Perfil de Docente.....	115
Figura 3.20 Entrenamiento.....	115
Figura 3.21 Reporte Diario.	116
Figura 3.22 Reportes por Fechas.....	116
Figura 5.1 Datos de entrada y salida de la hipótesis T.....	128
Figura 5.2 Comparación de caras.	129
Figura 5.3 Datos de entrada y s.	129
Figura 5.4 Salida de la hipótesis F.	130

ÍNDICE DE TABLAS

Tabla 1.1 Operacionalización de variable.	4
Tabla 1.2 Herramientas en el análisis y visualización de imágenes digitales.....	9
Tabla 2.1 Detalle de coeficientes de COCOMO II.	67
Tabla 2.2 Ecuaciones del Método COCOMO II.....	68
Tabla 3.1 Entradas de la red neuronal RBF.	84
Tabla 3.2 Entrenamiento de red neuronal.	86
Tabla 3.3 Resultado de red neuronal.	86
Tabla 3.4 Flujo de eventos a.	90
Tabla 3.5 Flujo de eventos b.	92
Tabla 3.6 Flujo de eventos c.	93
Tabla 3.7 Flujo de eventos d.	95
Tabla 3.8 Flujo de eventos e.	96
Tabla 3.9 Flujo de eventos f.	98
Tabla 3.10 Flujo de eventos g.	99
Tabla 3.11 Flujo de eventos h.	101
Tabla 3.12 Flujo de eventos i.	102
Tabla 3.13 Flujo de eventos j.	104
Tabla 3.14 Flujo de eventos k.	106
Tabla 3.15 Flujo de eventos l.	107
Tabla 3.16 Flujo de eventos m.	109
Tabla 4.1 Ponderación de funcionabilidad.	117
Tabla 4.2 Ponderación de usabilidad.	118

Tabla 4.3 Resultados de características de alto nivel ISO 9126.	120
Tabla 4.4 Coeficientes para los modos.	121
Tabla 4.5 Tabla Costo del sistema según líneas de código.	122
Tabla 4.6 Costo total etapas del Software.....	125
Tabla 4.7 Recursos de Hardware.....	126
Tabla 4.8 Recursos de Software de desarrollo.....	126
Tabla 4.9 De Insumos.	127
Tabla 4.10 Costo total del Proyecto.	127

1. MARCO PRELIMINAR

1.1 Introducción

La tecnología biométrica la cual es la ciencia que se dedica a la identificación de individuos a partir de una característica anatómica o un rasgo de su comportamiento. Busca obtener, clasificar y utilizar la información de estas características (anatómicas o de comportamiento) para conocer e identificar a las personas, restringir el acceso a sitios no permitidos, controlar los ingresos y salidas, autenticar información. Para esto se usa diferentes equipos electrónicos y métodos que desarrollan las mediciones biométricas y algoritmos que permiten digitalizar, clasificar y almacenar la información para poder utilizarla posteriormente.

Para la construcción del sistema se aplican redes neuronales, y en específico, la red base radial, en el cual presenta diferentes usabilidades. También se utilizará para su construcción MatLab (laboratorio de matrices) que es una herramienta de software matemático, un programa muy potente, que incluye herramientas de cálculo científico y técnico de visualización gráfica, así como un lenguaje de programación de alto nivel. Los cuales se utilizarán en la construcción de un sistema de reconocimiento facial, en el cual estará desarrollado de una forma amigable para que el usuario pueda interactuar con el sistema. El personal docente de la carrera Ingeniería Autotrónica, perteneciente a la UPEA, tomado distintas formas de control, desde las clásicas planillas de asistencia hasta la incorporación de sistemas biométricos de huella dactilar para el personal docente lo cual lleva a la necesidad de la construcción de nuevos métodos y sistemas más sofisticados de control de personal docente, por esas razones se plantea la construcción de nuevos métodos y sistemas más sofisticados de control de personal.

1.2 Planteamiento del Problema

Hoy la tecnología de reconocimiento facial está siendo utilizada en diferentes ámbitos como ser, el fraude de pasaportes, identificación de niños extraviados, soporte al orden público, como sistema de seguridad, minimiza el fraude en las identificaciones, sistemas de vigilancia y los sistemas biométricos.

En los últimos tiempos la gran necesidad de seguridad y vigilancia han llevado al desarrollo de distintos métodos y técnicas, las cuales han llevado a la creación de sistema biométricos cada vez más sofisticados, en la cual el sistema de reconocimiento facial va ganando una gran aceptación. Por ser un sistema experto que será entrenado para incrementar la habilidad de determinar, interpretar y organizar el contenido de la imagen o fotografía y encontrar un rostro en la misma el cual se procesará para la identificación de las personas en un lapso de tiempo corto.

Para lo cual, en la carrera de Ingeniería Autotrónica perteneciente a la Universidad Pública de El Alto, el control general del personal Docente, es realizado de manera manual, en partes diarios, que va generando el registro de ingresos y salidas de los funcionarios, el mismo que no presenta un grado de confiabilidad total.

1.2.1 Problema principal

La ineficiente forma de registro de planillas diarias de entradas y salidas del personal Docente de la carrera de Ingeniería Autotrónica que actualmente son realizadas manualmente, lo cual genera dificultades en los reportes de ingresos y salidas, generando así conflictos a la hora de control, diario y mensual de la asistencia para una buena toma de decisiones.

1.2.2 Problemas secundarios

- El registro manual de control de asistencia de los Docentes de la carrera Ingeniería Autotrónica, genera retardo al momento del reporte diario que se eleva al inmediato superior, para su verificación.

- Se genera reportes con errores de registro, las cuales son realizadas de forma manual.
- Dificultad en la realización de reportes diarios y mensuales de los Docentes, lo que ocasiona un trabajo moroso y tedioso.
- Se realiza un seguimiento académico inadecuado debido a la carencia de confiabilidad en el registro de control de entradas y salidas del personal docente.

Lo cual conlleva a plantear la siguiente interrogante a la que pretende responder con la presente Tesis:

¿Cómo realizar un control adecuado en el registro de entradas y salidas que permita generar un reporte ágil y oportuno del personal Docente?

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar un prototipo de Biometría facial basada en redes neuronales aplicando el algoritmo de Viola – Jones, para el control de ingreso y salida del personal Docente de la carrera Ingeniería Autotrónica.

1.3.2 Objetivo específico

- Analizar un diagnóstico de la forma de registro del personal Docente de la carrera Ingeniería Autotrónica.
- Desarrollar un modelo de registro de control de entradas y salidas.
- Realizar un reporte diario y mensual del personal Docente.
- Entrenar y diagnosticar la topología de RNA que sean adecuadas individualmente a cada Docente.

1.4 Hipótesis

El reconocimiento facial basada en redes neuronales aplicando el algoritmo de Viola – Jones, para el control de ingreso y salida del personal Docente de la carrera Ingeniería Autotrónica.

H1: El reconocimiento facial basada en redes neuronales aplicando el algoritmo de Viola – Jones, que permita mejorar el proceso de control de ingreso y salida del personal Docente de la carrera Ingeniería Autotrónica.

H0: El reconocimiento facial basada en redes neuronales aplicando el algoritmo de Viola – Jones, NO permitirá el proceso de control de ingreso y salida del personal Docente de la carrera Ingeniería Autotrónica.

1.4.1 Identificación de variables

Variable independiente

X: Control de reconocimiento facial al personal Docente de la carrera Ingeniería Autotrónica

Variable dependiente

Y: El resultado del reconocimiento de la imagen tomada mediante la cámara.

1.4.2 Operacionalización de variables

Tabla 1.1 Operacionalización de variable.

Variable independiente	Indicadores	Descripción	Actuador/ Instrumento
Control biométrico facial al personal Docente de la carrera Ingeniería Autotrónica.	<ul style="list-style-type: none"> • Redes neuronales (RNA). • Algoritmo de Viola – Jones. • Cámaras. 	La imagen tomada por la cámara y las características individuales de cada Docente.	<p>Técnica: Documental</p> <p>Instrumento: Ficha bibliográfica</p>
Variable Dependiente	Indicadores	Descripción	Actuador/ Instrumento

<p>El resultado del reconocimiento de la imagen tomada mediante la cámara.</p>	<ul style="list-style-type: none"> • Cámara biométrica para el control de asistencia del personal docente. • Reconocimiento facial. 	<p>El reconocimiento facial con base en un modelo de redes neuronales.</p>	<p>Técnica: Observación del reconocimiento de rostros en base a redes neuronales. Instrumento: Guía de observación</p>
--	---	--	--

Fuente: [Elaboración propia, 2020].

1.5 Justificación

1.5.1 Justificación científica

La tesis presentada es un aporte a la comunidad del conocimiento científico, mediante la aplicación de RNA – BR al reconocimiento fácil y mediante el mismo generar un “medio” del control de personal, con un nivel tecnológico acorde a nuestro medio.

1.5.2 Justificación técnica

Se utiliza por aplicar tecnología de redes neuronales y algoritmo de Viola – Jones, en que a la actualidad han superado las expectativas tanto en precisión como en calidad, para satisfacer el mejor control de registro del personal Docente.

1.5.3 Justificación económica

Una vez implantada el prototipo de investigación permitirá a la carrera de ingeniería de Autotrónica realizar el registro de entradas y salidas optimizando el costo de material de escritorios y la pérdida de los mismos

1.5.4 Justificación social

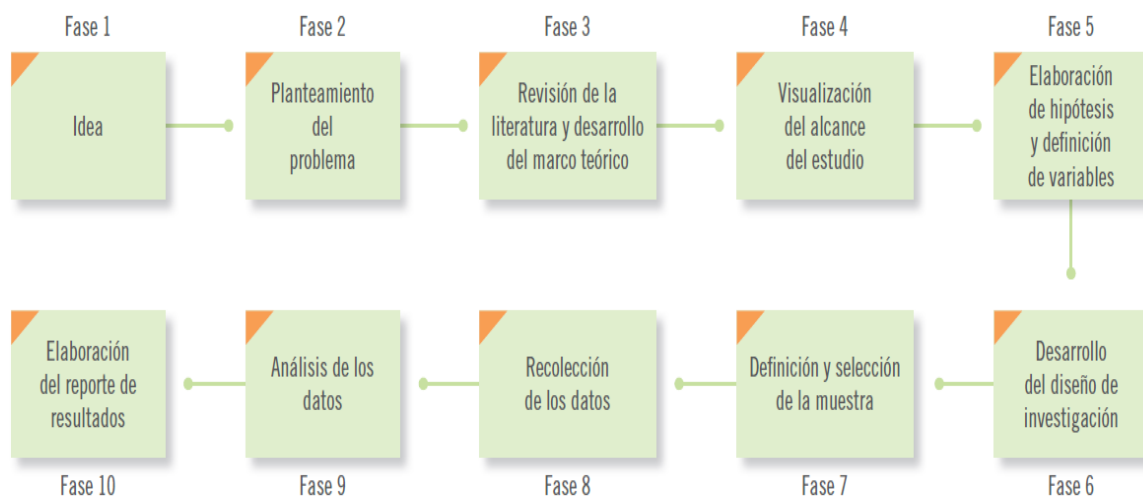
En nuestra sociedad se advierte poca responsabilidad en el horario de ingreso y salida en las diferentes fuentes de trabajo para lo cual se emplea diferentes métodos de control, y uno de los cuales más éxito tiene en ese objetivo son los sistemas de control biométrico. Mismas son utilizadas en diferentes ámbitos, ya gubernamentales, carreras universitarias, instituciones públicas y/o privados.

1.6 Metodología

1.6.1 Método científico

Según (Hernández, Fernández, Sampieri, 2010), La metodología de la investigación es concebida como un proceso constituido por diversas fases, interconectadas de una manera lógica y dinámica. La definición que adopta es la siguiente: La investigación es un conjunto de procesos sistemáticos y empíricos que se aplican al estudio de un fenómeno. Según esta definición se plantean las siguientes fases como se ve en la Figura N° 1.1:

Figura 1.1 Proceso de Investigación



Fuente: [Hernández, Fernández, Sampieri, 2010].

1.6.2 Método de ingeniería

1.6.2.1 Metodología mixta

La importancia de esta metodología es el construirse a partir de unir puntos más fuertes de las metodologías existentes en diferentes áreas.

Las características más importantes es la obtención de documentación que puede reemplazar parcialmente al experto, y servir a los diseñadores y/o usuarios como medio de documentación y referencia.

Las fases principales de la metodología mixta se destacan en cinco (5) etapas fundamentales, las cuales son:

1. Estudio preliminar:

- ✓ Definición del problema.
- ✓ Estudio de factibilidad.
- ✓ Análisis:
 - Posibilidad de la construcción.
 - Justificación de la construcción.

2. Desarrollo del prototipo:

- ✓ Licitación de conocimiento.
- ✓ Representación del conocimiento.
- ✓ Elección de la herramienta.
- ✓ Formalización: construcción de las reglas.
- ✓ Implementación del prototipo.

3. Desarrollo del sistema:

Los avances deben estar acompañados de evaluaciones internas de los expertos y la reconstrucción a partir del resultado de las evaluaciones.

4. Evaluación final:

El testeo final consiste en plantear casos nuevos y comparar la solución y procedimiento realizado por el experto y por el sistema.

5. Mantenimiento:

- ✓ Actualización del sistema.

1.7 Herramientas

1.7.1 MatLab R2018b

Es un lenguaje de alto nivel, que incluyen herramientas de cálculo numérico y visualización de imágenes. Es un programa de Mathworks orientado para realizar todo tipo de cálculos con vectores y matrices.

También presenta la posibilidad de realizar gráficos en dos y tres dimensiones. Es una herramienta potente que realiza cálculos complejos, con gran precisión, se emplea en aplicaciones en las que se trabajan con funciones matemáticas o matrices. Su entorno grafico es muy sencillo de aprender, lo que hace que sea uno de los lenguajes más empleados en diferentes ámbitos. MatLab tiene unas herramientas específicas para cada tipo de aplicación, llamadas toolboxes contienen una colección de funciones (M-files) que extienden el dominio de MatLab para resolver problemas de ámbitos particulares.

Tiene diferentes herramientas que son empleadas en diferentes áreas de trabajo como ser el proceso de señales y de imágenes, lógica difusa, los sistemas de control, etc. También permite realizar simulaciones del comportamiento de diferentes sistemas. Posee un Toolbox (SDC) para el procesado de imágenes que proporciona un conjunto de funciones y herramientas en el análisis y visualización de imágenes digitales, así como para el desarrollo de algoritmos.

Esta toolbox proporciona un conjunto flexible de herramientas para resolver problemas complejos en el tratamiento y el procesado de imágenes, para así poder extraer características específicas de esa imagen, las cuales son necesarias para el sistema biométrico a realizar.

Tabla 1.2 Herramientas en el análisis y visualización de imágenes digitales.

	Entorno grafico amigable	Software libre	Herramientas adicionales para el proceso digital de imágenes	Instalar complementos	Más información en el proceso de imágenes	Funciona en ventana de comando	Código fuente muy extendido
MatLab	✓		✓	✓	✓	✓	
Simulink	✓		✓	✓			

Fuente: [Elaboración propia, 2020].

1.7.2 Cámara web

Una cámara web necesita una computadora para transmitir las imágenes. Sin embargo, existen otras cámaras autónomas que únicamente necesitan un punto de acceso a la red informática, bien sea ethernet o inalámbrico. Para diferenciarlas de las cámaras web se las denomina cámaras de red. Ambas son útiles en tareas de seguridad, para video vigilancia.

1.7.3 XAMP

Es un servidor independiente multiplataforma, de software libre, que consiste principalmente en la base de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MariaDB, MySQL, PHP, Perl.

1.8 Límites y Alcances

1.8.1 Límites

Bajo el requerimiento de la carrera ingeniería Autotrónica, se tiene las siguientes:

- El sistema realizara reportes diarios y mensuales.
- El sistema no mostrará las asignaturas que son designadas para cada docente.

- El sistema no verificará los horarios de clases.
- El sistema no se acoplará con otros sistemas ya implantados como ser el caso se sistema de reconocimiento dactilar.

1.8.2 Alcances

El presente trabajo pretende realizar un sistema que como mínimo contenga el siguiente modulo:

- I. **Módulo de inicio de sesión:** consiste en la autenticación del administrador o usuario mediante un nombre de usuario y una contraseña.
- II. **Módulo de reconocimiento facial:** Consiste en el reconocimiento automático de un rostro en la imagen que ingresa al sistema, para poder de esa manera, extraer las características necesarias para ser comparadas con la base de datos y registrar los ingresos y salidas del Personal Docente.
- III. **Modulo Gestión de Usuario:** Lista a los administradores autorizados para el manejo del sistema, provisto con un nombre de usuario y una contraseña.
- IV. **Modulo Docente:** Registra un nuevo docente, despliega el perfil de docente y realiza el entrenamiento de la red neuronal de base radial (RNA - RBF).
- V. **Módulo de Reportes:** Se generará los reportes diarios, mensuales, y todo el aprovechamiento que se le dará al sistema de control de registros del personal Docente.

1.9 Aportes

El estudio permite integrar el flujo de información concerniente a las imágenes, brindando el reconocimiento y de esa manera se automatice las funciones que se realizan, brindando una interface de pantallas amigables y fácil manejo al usuario.

Para ello se utiliza el desarrollo de menús de selección, siendo la implementación del sistema propuesto, que se logra a partir del análisis, diseño codificación y pruebas de calidad del mismo, para finalmente lograr una herramienta capaz de manejar y brindar información precisa e integrada.

2. MARCO TEÓRICO

2.1 Ingeniería de Requerimientos

Los requerimientos es una etapa muy importante para poder comprender las necesidades del sistema que se desea desarrollar.

Para lo cual se utilizará redes neuronales artificiales de Función Base Radial (RBF), para el reconocimiento facial. El mismo que sea capaz de memorizar un conjunto de rostros de diferentes personas, donde cada persona dispondrá de un conjunto determinado de imágenes, así mismo se deberá verificar ciertos aspectos, tal como el formato y la resolución de las imágenes faciales basadas en niveles de gris. Con lo cual se podrá realizar la tarea de identificación de la persona.

Analizando el problema y observando lo expuesto, se describirán los siguientes requerimientos para poder realizar el sistema.

2.1.1 Requerimientos funcionales

Se describe lo que debe realizar el Sistema.

- El sistema debe tomar una imagen y procesarla, cuando la persona se acerque a la cámara web.
- La imagen tomada, será procesada para segmentar el rostro que se encuentra en ella.
- La imagen tomada debe ser segmentada de forma automática.
- La imagen segmentada, será procesada, para convertirla en una imagen de niveles de grises, hasta llegar a una imagen con los ojos, nariz y boca marcada y reconocida.
- Cargar a la memoria almacenada de las imágenes de las personas, previamente registradas en el sistema.
- Configurar los parámetros de las RNAs. de RBF.
- Entrenar la red neuronal artificial con las imágenes de los rostros almacenados en la base de datos del sistema.

- Mostrar el resultado del entrenamiento y tiempo empleado por cada red para lograr el aprendizaje.
- Memorizar el entrenamiento en la red neuronal artificial RBF, y actualizarlo si se obtiene un mejor resultado.
- Reconocer (identificar), la imagen del rostro entrante, mostrando el nombre completo y su número de identificación, de cada usuario.

2.1.2 Requerimientos no funcionales

Se describen las cualidades generales que ha de tener el sistema.

- La estructura y diseño del sistema deben adaptarse fácilmente a cualquier cambio o mejora en la misma.
- El sistema debe poseer una interfaz amigable, para el usuario.

2.1.3 Requisitos de Software

Se describe los requerimientos mínimos de software que debe cumplir el computador, para así poder soportar y ejecutar el sistema.

- Sistema Operativo Windows 10 (o posterior).
- MatLab R2018b.
- XAMPP 7.3.12

2.1.4 Requisitos de Hardware

Se describen los requerimientos mínimos de hardware que debe cumplir el computador para poder ejecutar y soportar el sistema.

- Procesador Intel® Core™ i3-8320
- 8 MB Intel® Smart Cache Caché
- 4 Núcleos.
- 8 Subprocesos.
- 4.60 GHz Frecuencia turbo máxima.
- 8th Generation.
- 8 GB de memoria principal.

- 20 GB de disco duro.
- Monitor de 15” (o posterior)
- Cámara web HD.
- Impresora

2.2 Teoría Sobre RNA – Base Radial

2.2.1 Redes neuronales artificiales (RNA)

Las redes neuronales artificiales han tenido un estudio intensivo en los últimos años, lo que originó la maduración de las primitivas ideas desarrolladas en los años sesenta. El procesamiento de la información realizado por estos sistemas es un tipo alternativo al que utilizan los actuales sistemas informáticos, los cuales procesan secuencialmente instrucciones almacenadas en una memoria y manipulan datos de la misma memoria.

En las redes neuronales en cambio, el paradigma emula el modelo biológico de aprendizaje y computación, con una fuerte base de procesamiento en paralelo, y estando involucrados en cada unidad de cálculo el proceso y la memoria [N & Gomez, 2001].

Las redes neuronales artificiales están inspiradas en el modelo de neurona biológica, usando unidades elementales de procesamiento que mimetiza algunas de las características de las neuronas biológicas. [Gutiérrez 2002].

El comportamiento global de una red determina su capacidad para ensayar hipótesis, detectar patrones estadísticos y regularidades o ajustar dinámicamente un modelo implícito implementado en la misma arquitectura. Este comportamiento va más allá de la suma de las potencialidades de las neuronas que la componen. El resultado es la emergencia de propiedades nuevas que pertenecen al sistema como un todo [Gutiérrez 2002].

2.2.1.1 Definiciones básicas de RNA

Existen diferentes definiciones que pueden ayudarnos y ser aclaratorias:

- “Una red neuronal es un procesador distribuido y con estructura paralela que tiene una tendencia natural a almacenar conocimiento experimental, haciéndolo apto para su uso. Se parece en dos cosas al cerebro: el conocimiento es adquirido por la red a través de un proceso de aprendizaje y este conocimiento se almacena en las pesas sinápticos o conexiones entre neuronas”. [López, 2012]
- “Una red neuronal es un modelo computacional con un conjunto de propiedades específicas, como son la habilidad de adaptarse o aprender, generalizar u organizar la información, todo ello basado en un procesamiento eminentemente paralelo”. [Caballero, 2004].
- “Los sistemas neuronales artificiales o redes neuronales son sistemas celulares físicos capaces de adquirir, almacenar y utilizar conocimiento extraído de la experiencia” [López, 2012].

2.2.2 Elementos básicos de procesado

Estos elementos básicos de procesado son los encargados de realizar las operaciones de la red neuronal. Se presentarán a continuación de manera ordenada dichos elementos constitutivos de las redes [N & Gomez, 2001].

2.2.2.1 Sistema de patrones entrada – salida

Se llamará patrón de entrada de una RNA al conjunto de variables independientes de entrada que se han elegido para representar el sistema que se quiere modelar. En el caso de tratarse de datos sin evolución temporal, pertenecen a un instante concreto y se representan por un vector de dimensión R , siendo R el número de variables de entradas independientes. Un sistema de patrones de entradas será el conjunto de todos los patrones de entradas que formarán el grupo de datos disponibles para el entrenamiento, y que será representado como una matriz de dimensión $R \times m$, siendo m el número de datos. [N & Gomez, 2001].

Del mismo modo, un patrón de salida es el conjunto de variables de respuesta del sistema a modelar, correspondiente a un patrón de entrada. Un patrón de salida tiene dimensión $S \times 1$ (siendo S el número de variables de salida independientes que se

quieren obtener) y se define el sistema de patrones de salida (de dimensión $S \times m$, donde m es el mismo número que para las entradas) al conjunto de todos los patrones de salida disponibles [N & Gomez, 2001].

Este sistema de patrones entrada-salida se utilizará para el entrenamiento y la posterior validación de la red. El objetivo de estas dos fases será ajustar las funciones de las RNA de manera que al aplicarlas sobre los patrones de entrada se obtengan resultados lo más parecido posible a los patrones de salida $S \times m$, [N & Gomez, 2001].

2.2.2.2 Conexiones y pesos sinápticos

Las conexiones sinápticas son las vías o caminos de comunicación entre los diferentes elementos de procesado (neuronas) y entre éstos y las entradas/salidas del sistema. Las conexiones se establecen únicamente entre dos elementos y solo pueden transmitir información en un sentido. La mayoría de las veces estas conexiones también procesan la información que transmiten de un modo sencillo. Cada conexión lleva asociado un peso o ponderación, que en adelante denotaremos con w_{ji} , donde el índice i corresponde al número de neuronas de la que parte la conexión y j a la que llega [N & Gomez, 2001].

Además de los pesos propiamente dichos, específicos para las conexiones que unen las neuronas, existen unos pesos adicionales a considerar llamados sesgos, umbrales o vías y que representaremos con b . Estos sesgos están asociados a unas conexiones con origen en una unidad ficticia de entrada y la neurona de llegada. La inclusión de este grado de libertad mejora la convergencia del ajuste de los pesos durante el proceso de entrenamiento, ya que evita la restricción de que la solución pase por el origen [N & Gomez, 2001].

2.2.2.3 Funciones de activación

Las neuronas biológicas pueden estar excitadas o no excitadas, con lo que poseen cierto “estado de activación”. Las neuronas artificiales también tienen grados de activación, que pueden limitarse únicamente a dos, como en el caso de las biológicas, o variar entre un rango de valores (normalmente definidos entre $[0, 1]$ o entre $[-1, 1]$).

Este valor indica el estado de la neurona: inactiva (0 o -1), activa (1), o bien un estado intermedio entre estos límites que indica su grado de activación.

Para obtener la salida final de una neurona se debe aplicar a la entrada ponderada total una función de activación que calcula el valor de salida o estado de actividad de la neurona.

Las neuronas de la capa de salida suelen utilizar funciones de activación identidad (función lineal), mientras que para los demás elementos de procesado se suelen utilizar funciones no lineales. Estas no linealidades pueden ser modeladas con funciones de tipo escalón, cuya respuesta es más fiel a la respuesta biológica, o con sigmoides, cuyo comportamiento matemático es menos hostil.

Aunque existen un gran número de funciones de activación diferentes, las de uso más común son las funciones sigmoideas, ya sea en su forma binaria (o logarítmicas) o en su expresión bipolar (o tangente hiperbólica). Sus ecuaciones se muestran a continuación y sus representaciones gráficas se presentan en la

Figura 2.1 Sigmoidea.

Sigmoidea Binaria:	$f1(x) \frac{1}{1+e^{-x}}$	(2.1)
Sigmoidea Bipolar:	$f2(x) \frac{1-e^{-x}}{1+e^{-x}}$	(2.2)

Fuente: [N & Gomez, 2001].

El parámetro 'x' en las ecuaciones 2.1 y 2.2 es siempre positivo y controla la pendiente de la función sigmoidea. Los cambios en esta pendiente influyen de forma directa en la brusquedad de los cambios a la salida de la neurona: mientras que para valores pequeños de 'x' la variación es casi lineal, para valores elevados casi se obtienen respuestas tipo escalón, con lo que no hay casi valores intermedios entre las asíntotas de saturación. La diferencia entre las dos funciones sigmoideas es el rango de variación de cada una de sus salidas: mientras que la sigmoidea binaria está comprendida entre [0, 1], la bipolar lo está entre [-1, 1], [N & Gomez, 2001].

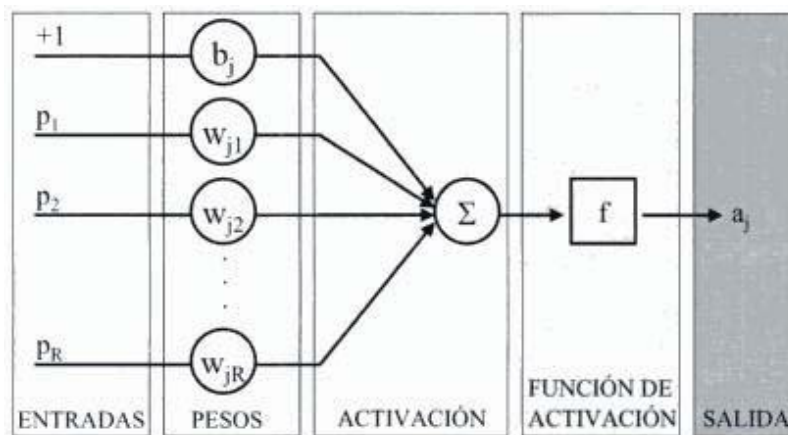
Un requisito que han de cumplir las funciones de activación para poder aplicar los algoritmos de entrenamiento tipo Backpropagation, es que sean derivables, condición que cumplen todas las funciones de activación descritas [N & Gomez, 2001].

2.2.2.4 Elemento de procesado: neurona

La neurona es el elemento que procesa la información en la RNA y juega un papel fundamental en su funcionamiento. El modo de operación de las neuronas es el siguiente:

El conjunto de entradas que llegan a la neurona j (p_i), será multiplicado por cada uno de los pesos (w_{ij}), asociados a las conexiones. Se realizará una suma de esta ponderación de entradas con los pesos, junto con el valor del sesgo de la neurona (b_j). Una vez obtenida esta suma ponderada, se obtiene la salida final al aplicar una función de activación propia de la neurona [Serrano, Soria, Martin, 2010].

Figura 2.2 Estructura y funcionamiento de una neurona artificial.



Fuente: [serrano, Soria, Martin, 2010].

2.2.2.5 Asociación de neuronas

Normalmente las neuronas se organizan en distintos niveles de la red neuronal, que suelen denominarse capas. Estas capas se clasifican según su situación dentro de la red siguiendo el siguiente criterio:

- **De entrada:** Es siempre la primera capa de una RNA, y tiene tantas neuronas como entradas a la red. Cada neurona tiene una sola entrada que proviene directamente del exterior y varias salidas a neuronas de capas posteriores.

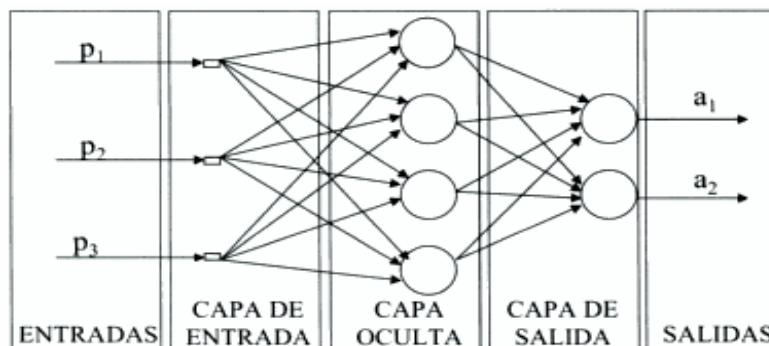
A menudo, en el recuento de capas de una RNA no se incluye la de entrada, ya que sus neuronas se limitan a distribuir las entradas externas a las capas posteriores, pero no realizan ninguna operación “típica” de estos elementos de procesado [Serrano, Soria, Martín, 2010].

- **Ocultas:** Estas neuronas son las que realizan el trabajo y no tienen contacto directo con las señales externas a la red neuronal artificial. Las neuronas de estas capas son las que mejor se ajustan a la definición de neuronas que se ha hecho en apartados anteriores. Puede haber varias capas de neuronas ocultas, desde cero hasta un número elevado.

La primera de las capas ocultas estará conectada con la capa de entrada y la última de estas capas comunicará generalmente con la capa de salida. La forma de interconexión entre las distintas capas ocultas junto con el número de neuronas en cada capa determinará la topología de la red neuronal artificial [Serrano, Soria, Martín, 2010].

- **De salida:** es la capa de neuronas que transmite la información generada por la red al exterior. El número de neuronas de esta capa será igual al número de salidas de la RNA.

Figura 2.3 Asociación de neuronas por capas.



Fuente: [Serrano, Soria, Martín, 2010].

La figura 2.3 se muestra un ejemplo de RNA multicapa con 3 entradas, 2 salidas y una capa oculta de 4 neuronas. En este caso específico, cada capa está totalmente interconectada con la siguiente, es decir, la salida de una neurona de una capa anterior llega a la entrada de todas las neuronas de la capa siguiente, [serrano, Soria, Martin, 2010].

La clasificación de neuronas en capas en un primer paso para introducir las diferentes topologías o arquitecturas que se pueden encontrar en las RNA, y que se describirán a continuación, [serrano, Soria, Martin, 2010].

2.2.3 Arquitecturas de RNA

La organización de las neuronas y el tipo de conexiones que existen entre ellas es lo que se denomina arquitectura de las RNA. La elección de la arquitectura óptima para cada aplicación específica a la que se quiera aplicar las RNA es un tema clave para encontrar una solución satisfactoria al problema a tratar.

Un posible esquema de la clasificación de las RNA en función de dichos parámetros aparece en el cuadro 2.1. Como se puede observar en el cuadro, los parámetros fundamentales que determinan la arquitectura de la RNA son el número de capas y la conectividad que se establece entre las neuronas. Tanto la nomenclatura de cada una de las RNA, como las diferentes subdivisiones dentro de estos dos criterios generales, se trata a continuación, [serrano, Soria, Martin, 2010].

Figura 2.4 Esquema de clasificación de una RNA.

CAPAS		CONECTIVIDAD		
Numero de capas	Numero de neuronas	Tipo de conexiones	Tipo de información	Grado de conectividad
Monocapa	1	Hacia adelante	Estatica	Parcial
Multicapa	2	Lateral	Dinamica	Total
	N neuronas	Con Retardo		

Fuente: [Elaboración propia, 2020].

Existen dos tipos de clasificaciones de las RNA según su número de capas y según su conectividad:

2.2.3.1 Clasificación según número de capas

salida, pero también existen casos donde las conexiones entre las neuronas son más complicadas.

Las redes monocapa se utilizan típicamente en tareas relacionadas con lo que se conoce como auto-asociación; por ejemplo, para regenerar informaciones de entradas incompletas o distorsionadas que se presentan a la red, o para el reconocimiento de patrones. Por otra parte, los problemas que pueden ser resueltos por una red monocapa son bastante limitados, teniendo que recurrir en la mayoría de los casos a las redes multicapa. Ejemplos de redes monocapa son las redes de Hopfield o la máquina de Boltzmann, [serrano, Soria, Martin, 2010].

Redes multicapa. Las redes multicapa están compuestas por al menos una capa oculta entre la capa de entrada y la de salida de la RNA.

Las redes multicapa pueden resolver problemas más complicados que las redes monocapa, a cambio eso sí, de tratar con superficies más complejas y un entrenamiento mucho más costoso y con más dificultades para evitar mínimos locales. Normalmente la dificultad del problema a resolver va asociada con el tamaño de las redes multicapa, que está definido por el número de capas ocultas y de neuronas en cada una de ellas. Como ya se ha comentado, para obtener una mejor generalización se tiene que determinar apropiadamente el tamaño de las redes a usar en cada caso, [serrano, Soria, Martin, 2010].

2.2.3.2 Clasificación según tipo de conectividad

La conectividad entre las neuronas canaliza la información de salida de una neurona como entrada de otra. Se presentan a continuación los diferentes tipos de conexiones entre las neuronas en una red neuronal, [Enrique C., 2004].

Según tipo de conexiones:

- **Conexiones hacia delante:** el sentido de transmisión de la información en este tipo de conexiones se realiza desde neuronas de capas inferiores (más cercanas a la capa de entrada) hacia neuronas de capas superiores (más cercanas a la salida). También se denominan feedforward, [Enrique C., 2004].
- **Conexiones laterales:** son las conexiones que se hacen entre neuronas de una misma capa. Estas conexiones se diseñan como excitadoras, permitiendo la cooperación (con peso positivo) o la inhibición (con peso negativo) entre neuronas, lo que establece una competición entre neuronas de la misma capa, [Enrique C., 2004].
- **Conexiones con retardo:** Los elementos de retardo se incorporan en las conexiones para implementar modelos dinámicos temporales (redes dinámicas). Se almacenan los datos antiguos para después procesarlos junto con la información actual, [Enrique C., 2004].

Según tipo de información:

La última clasificación que se ha considerado es la del tipo de información que reciben las redes y se presenta a continuación:

- **Redes estáticas.** El tipo de información que manejan las redes estáticas tiene carácter estacionario, es decir, el valor de la salida solo depende del valor de las entradas en el mismo momento en el que ésta se produce.

Estas redes se pueden caracterizar estructuralmente por la inexistencia de bucles de realimentación y de elementos de retardo entre los distintos elementos de proceso que las forman. Normalmente van asociadas a las redes con conexiones hacia delante (feedforward), [Enrique C., 2004].

- **Redes dinámicas.** La información que se maneja en este tipo de redes incluye la variable tiempo, es decir, la salida depende de las variables de entrada en el momento actual y de los valores de entrada y/o salida en momentos anteriores.

Estas redes suelen asociarse al tipo de redes recurrentes explicadas con anterioridad.

Para incluir la variable tiempo, se puede incorporar en la red retardos temporales o conexiones de tipo feedbackward entre las capas de neuronas.

El entrenamiento para las redes dinámicas puede plantear problemas de convergencia y estabilidad, y los resultados que generan son en general de difícil análisis, [Enrique C., 2004].

2.2.4 Entrenamiento de RNA

Como se ha ido explicando en los apartados anteriores, los datos de entrada de la RNA se procesan mediante un conjunto de **operaciones** internas, que proporcionan una salida de la red. Las secciones anteriores se han centrado en caracterizar las redes desde el punto de vista de estructura y funcionamiento matemático interno, [Enrique C., 2004].

Desde el punto de vista de la aplicación de las RNA a un problema específico, existen dos fases: la fase de entrenamiento o aprendizaje y la fase de testeo o de prueba. En la fase de entrenamiento, se usa un conjunto de datos o patrones de entrenamiento para determinar las características que definen el modelo neuronal. Una vez entrenado este modelo, se pasa a la llamada fase de testeo, en la que se procesan nuevos patrones no presentados a la RNA hasta el momento y se analizan la capacidad de generalización de la misma, [Enrique C., 2004].

Puesto que la topología de la red y las funciones de activación son características fijas de la RNA, los cambios que se producen durante el proceso de entrenamiento se reducen a la modificación de los pesos asociados a las conexiones entre las neuronas, [Enrique C., 2004].

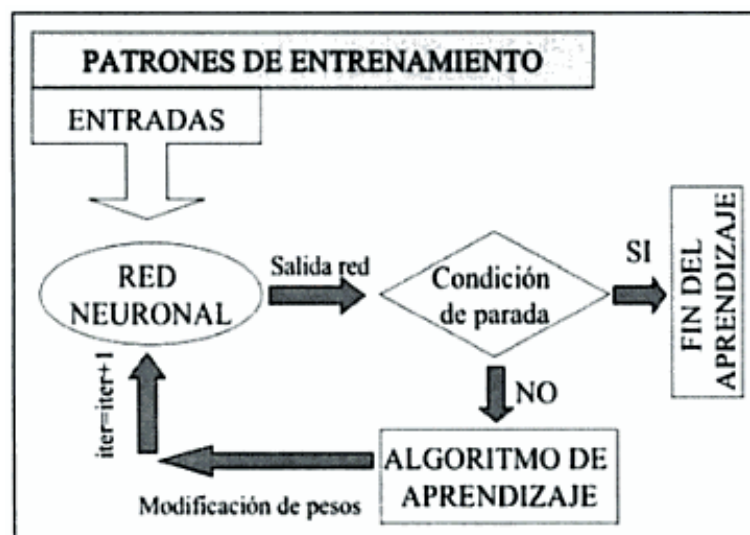
Existen distintos tipos de criterios para modificar los pesos de las RNA, que se pueden agrupar en dos estrategias generales de aprendizaje: entrenamiento supervisado y no supervisado, y que se detallan a continuación: [Enrique C., 2004].

2.2.4.1 Entrenamiento no supervisado

Tal y como se puede ver en la figura 2.4, en el entrenamiento no supervisado el conjunto de datos de entrenamiento solo está formado por los datos de entrada y no se dispone de los datos de salida de estos patrones. Se suele decir que la red es entrenada sin maestro, ya que ésta no recibe ninguna información del exterior que le indique si la respuesta generada es o no correcta. Las redes que usan este tipo de entrenamiento también se denominan auto-organizativas. La red aprende a adaptarse basándose en las experiencias recogidas de los patrones de entrenamiento vistos con anterioridad, encontrando regularidades, correlaciones o categorías entre estos.

En definitiva, la red modifica los pesos de forma que los vectores de entrada más similares sean asignados a la misma unidad de salida (o clúster), en lo que se conoce como mapeo de características. De ello se deduce que la aplicación fundamental de este tipo de entrenamiento es la clasificación o identificación de patrones de entrada. Ejemplos de redes con este tipo de entrenamiento son los mapas auto-organizativos de Kohonen. Las principales clases de entrenamientos no supervisados son el aprendizaje de Hebb y el aprendizaje competitivo y cooperativo, [serrano, Soria, Martin, 2010].

Figura 2.5 Diagrama de flujo de entrenamiento no supervisado.



Fuente: [Serrano, Soria, Martin, 2010].

2.2.4.2 Entrenamiento supervisado

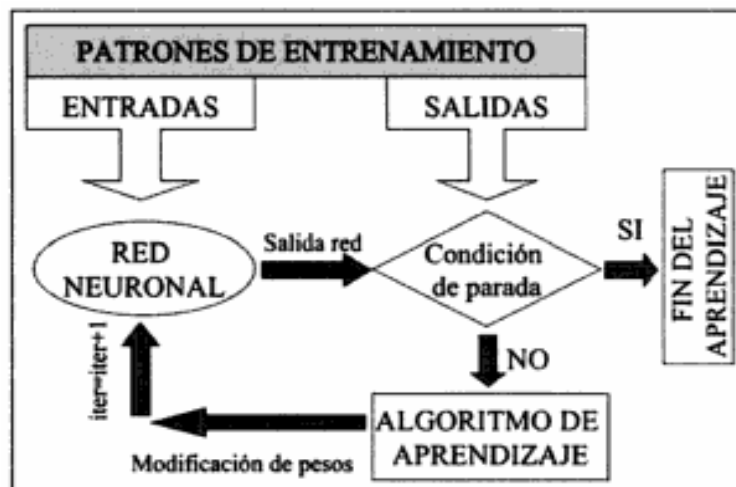
A diferencia del caso anterior, y tal y como se muestra en la figura 2.5, para este tipo de aprendizaje se dispone de unos patrones de entrenamiento compuestos por un conjunto de pares de elementos entrada-salida.

En este caso, el proceso de aprendizaje se realiza mediante un entrenamiento controlado por un agente externo (supervisor, maestro) que determina la respuesta que debería generar la red a partir de una entrada determinada. En cada iteración, el supervisor comprueba la salida de la red y en el caso de que ésta no coincida con la deseada, se procede a modificar los pesos de las conexiones con el fin de conseguir que la salida obtenida se aproxime más a la deseada. La red realizará varias iteraciones modificando los pesos de sus conexiones hasta alcanzar la condición de paro del entrenamiento, dejando a partir de estos momentos los pesos fijos. Estos pesos formarán el modelo de RNA que se usará en la fase de testeo que se ha comentado anteriormente.

Dentro de esta clasificación también existen diferentes formas de implementación:

- **Entrenamiento por refuerzo.** Se aplica en casos en los que no se conoce o no se desea indicar exactamente las salidas deseadas para los patrones de entrada.

Figura 2.6 Diagrama de flujo de entrenamiento supervisado.



Fuente: [serrano, Soria, Martin, 2010].

- **Entrenamiento por corrección de error.** Consiste en ajustar los pesos de las conexiones de la red en función de la diferencia entre los valores deseados y los obtenidos en la salida de la red, con el objetivo de minimizar esta diferencia.

Dentro del tipo de entrenamiento por corrección del error existe una gran variedad de algoritmos utilizados para las RNA, normalmente asociados a la tipología concreta de red utilizada. Por ejemplo, la regla de entrenamiento del perceptrón utiliza el ajuste de pesos. Esta regla, formulada por Rossemblatt en 1948, fue una de las primeras reglas implementadas en las redes neuronales, sólo es aplicable a redes de una sola capa.

Posteriormente surgieron algoritmos más evolucionados, que proporcionaban un entrenamiento más rápido y un campo de aplicación más amplio. Un ejemplo es la regla de aprendizaje Delta o regla del mínimo cuadrado, que a diferencia de la regla anterior considera todas las neuronas anteriores a la salida, con lo que se dispone del error de la red global, que se distribuye después entre las conexiones predecesoras.

A raíz de la regla Delta, surgió la regla Delta generalizada, más conocida como algoritmo de Backpropagation, que permite aplicar la regla delta a redes multicapa ya que actualiza los pesos de las neuronas de las capas ocultas. Este algoritmo ha dotado a las RNA de un extenso campo de aplicación y es ampliamente utilizado actualmente en las aplicaciones con RNA, [serrano, Soria, Martin, 2010].

2.2.5 Redes neuronales artificiales de base radial

Las redes de neuronas de base radial son redes multicapa con conexiones hacia adelante, al igual que el perceptrón multicapa. Las redes de base radial se caracterizan porque están formadas por una única capa oculta y cada neurona de esta capa posee un carácter local, en el sentido de que cada neurona oculta de la red se activa en una región diferente del espacio de patrones de entrada. Este carácter local viene dado por el uso de las llamadas funciones de base radial, generalmente la función gaussiana, como funciones de activación. Las neuronas de la capa de salida de las redes de base radial simplemente realizan una combinación lineal de las activaciones de las neuronas ocultas.

La mayor contribución a la teoría, diseño y aplicaciones de las redes de neuronas de base radial se debe a Moody y Darken, Renals y a Poggio y Girossi. Uno de los objetivos iniciales de los autores era construir una red de neuronas que requiriese un menor tiempo de aprendizaje que el que necesita el perceptrón multicapa y, de este modo, disponer de una red de neuronas que pudiera ser apropiada para aplicaciones a tiempo real. Esto se consiguió incorporando funciones de activación locales en las neuronas ocultas de la red, lo cual permitía que sólo unas pocas neuronas ocultas tuvieran que ser procesadas para nuevos patrones de entrada. Al igual que el perceptrón multicapa, las redes de neuronas de base radial son aproximadores universales, en el sentido de que pueden aproximar cualquier función continua sobre un compacto de \mathbb{R}^n .

Las funciones de base radial definen hiper esferas o hiper elipses que dividen el espacio de entrada. Por tanto, cada neurona oculta de la red de base radial construye una aproximación local y no lineal en una determinada región de dicho espacio.

Puesto que la salida de la red es combinación lineal de las funciones de base radial, las aproximaciones que construyen las redes de base radial son combinaciones lineales de múltiples funciones locales y no lineales. De este modo, se suele decir que las redes de base radial aproximan relaciones complejas mediante una colección de aproximaciones locales menos complejas, dividiendo el problema en varios subproblemas menos complejos. Esto hace que las aproximaciones construidas por las redes de base radial sean de naturaleza diferente a las aproximaciones globales y basadas en hiper planos que construye el perceptrón multicapa, sin embargo, cada una de estas clases de aproximaciones – redes de base radial y perceptrón multicapa – tiene sus propias características.

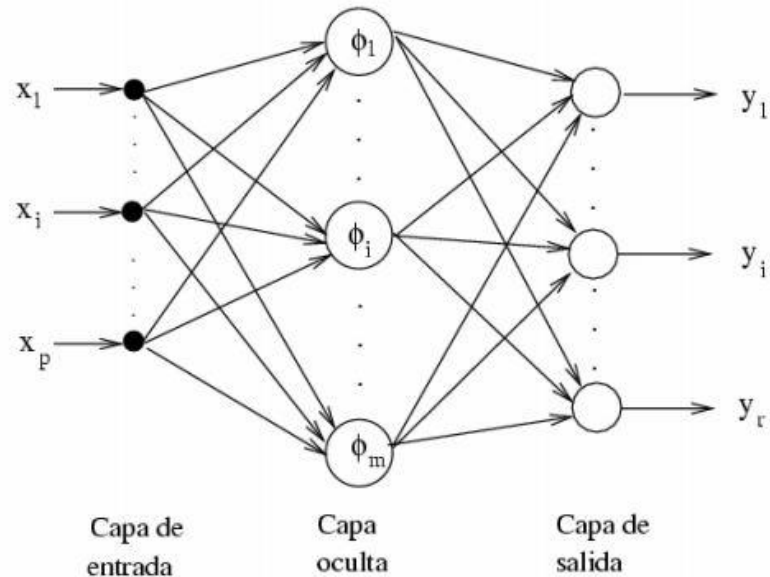
Las redes de neuronas de base radial han sido aplicadas a una gran variedad de problemas, aunque es necesario señalar que su aplicación no ha sido tan extendida como el caso del perceptrón multicapa, sin embargo, se han utilizado en diferentes campos, como análisis de series temporales, procesamiento de imágenes, diagnósticos médicos, reconocimiento automático del habla, etc. [Serrano, Soria, Martín, 2010].

2.2.5.1 *Arquitectura de las redes de base radial*

Las redes de neuronas de base radial están formadas por tres capas de neuronas: la capa de entrada, una única capa oculta y la capa de salida. La capa de entrada la componen un conjunto de neuronas que reciben las señales del exterior, transmitiéndolas a la siguiente capa sin realizar ningún proceso sobre dichas señales. Las neuronas de la capa oculta reciben las señales de la capa de entrada y realizan una transformación local y no lineal sobre dichas señales. Este carácter local es lo que la diferencia del perceptrón multicapa, no solo en cuanto a arquitectura, sino también en cuanto a comportamiento. Esta capa es la única que incluye componentes no lineales en las redes de base radial. Y, finalmente, la capa de salida que realiza una combinación lineal de las activaciones de las neuronas ocultas, que actúa además como salida de la red, [Gonzales, 2016]

Las redes de neuronas de base radial son redes con conexiones hacia delante, como se observa en la figura 2.6, y estas conexiones se dirigen siempre de una capa a la siguiente capa. La red se caracteriza porque las conexiones de la capa de entrada a la capa oculta no llevan asociado ningún peso, mientras que, y como es habitual, en el contexto de redes de neuronas, las conexiones de la capa oculta a la capa de salida sí llevan asociado un número real o peso de la conexión. En lo referente a los umbrales de las neuronas, en las redes de base radial únicamente las neuronas de salida poseen un umbral, que también se suele tratar al igual que en el perceptrón multicapa como una conexión más de la neurona cuya entrada es constante e igual a 1, [Gonzales, 2016].

Figura 2.7 Arquitectura de la red de neuronas de base radial.



Fuente: [Gonzales, 2016].

Las redes de neuronas de base radial definen una relación no lineal entre las variables de entradas y las variables de salida de la red, propagando hacia las salidas las señales o muestras recibidas en la entrada.

2.2.5.2 Activación de las neuronas de la red de base radial

Dada una red de neuronas de base radial con p neuronas en la capa de entrada m neuronas en la capa oculta y r neuronas en la capa de salida, las activaciones de las neuronas de salida para el patrón de entrada n , $X(n)=(x_1(n), x_2(n), \dots, x_p(n))$, denotadas como $y_k(n)$, vienen dadas por la siguiente expresión: [Gonzales,2016].

$$y_k(n) = \sum_{i=1}^m w_{ik} \phi_i(n) + u_k \quad \text{Para } k=1,2,\dots,r \quad (2.3)$$

Donde w_{ik} representa el peso de la conexión de la neurona oculta i a la neurona de salida k , u_k es el umbral de la neurona de salida k , $\phi_i(n)$ son las activaciones de las neuronas ocultas para el patrón de entrada $X(n)$. Se observa en la ecuación 2.3 que las neuronas de salida de la red utilizan la función de activación identidad, realizando

una transformación lineal de las activaciones de todas las neuronas ocultas, [Gonzales, 2016].

Las funciones univariada ϕ_i , también conocidas como funciones de base radial, determinan las activaciones de las neuronas ocultas de la red en función del vector de entrada a la red $X(n)$ y vienen dadas por la siguiente expresión:

$$\phi_i(n) = \phi \frac{\|X(n) - C_i\|}{d_i} \quad \text{Para } i = 1, 2, \dots, m \quad (2.4)$$

Donde ϕ es una función de base radial; $C_i = (C_{i1}, \dots, C_{ip})$ son vectores que representan los centros de la función de base radial; d_i son números reales que representan la desviación, anchura o dilatación de la función de base radial; y $\|X(n) - C_i\|$ es la distancia euclidiana del vector de entrada $X(n)$ al centro C_i , definida como:

$$\|X(n) - C_i\| = \left(\sum_j^p (x_j(n) - c_{ij})^2 \right)^{\frac{1}{2}} \quad (2.5)$$

Por lo tanto, la activación de una neurona oculta en las redes de base radial depende de la distancia del patrón de entrada $X(n)$ al centro C_i de la función de base radial.

Estas funciones bases ϕ poseen un carácter local, pues son funciones que alcanzan un nivel cercano al máximo de su recorrido cuando el patrón de entrada $X(n)$ está próximo al centro de la neurona; a medida que el patrón se aleja del centro, el valor de función va tendiendo al valor mínimo de su recorrido, como se puede apreciar en la ecuación 2.4.

La función de base radial puede adoptar diferentes formas y expresiones entre ellas:

- Función gaussiana:

$$\phi(r) = e^{\left(\frac{-r^2}{2}\right)} \quad (2.6)$$

- Función inversa cuadrática:

$$\phi(r) = \frac{1}{1+r^2} \quad (2.7)$$

- Función inversa multicuadrática:

$$\phi(r) = \frac{1}{\sqrt{1+r^2}} \quad (2.8)$$

En el contexto de redes de neuronas de base radial, la más utilizada es la función gaussiana. Por tanto, la activación de las neuronas ocultas de las redes de base radial viene dada, generalmente, por la siguiente expresión:

$$\phi(r) = \exp\left(-\frac{\|x(n)-c_i\|^2}{2d_i^2}\right) = \exp\left(-\frac{\sum_{j=1}^p (x_j(n)-c_{ij})^2}{2d_i^2}\right) \quad \text{Para } i=1,2,\dots,m \quad (2.9)$$

Las salidas de las redes de neuronas de base radial (ecuación 2.5) son, por tanto, una combinación lineal de gaussianas, cada una de las cuales se activa para una determinada porción del espacio definido por los patrones de entrada, [Gonzales,2016].

Carácter local de las redes de base radial

Las funciones de base radial ϕ se caracterizan porque poseen un nivel máximo de activación para valores de entrada cercanos a cero y dicho nivel decrece a medida que la variable de entrada se aleja de dicho punto.

En la ecuación 2.9 se observa que la activación de la neurona oculta i en una red de base radial viene dada por la función gaussiana de centro C_i y desviación d_i . Por tanto, si el patrón de entrada a la red $X(n)$ está en la vecindad del centro C_i , la neurona oculta i alcanzará un valor alto de activación. A medida que el patrón se aleja del centro dependiendo de la desviación la activación de la neurona disminuye, y puede activarse otra neurona oculta de la red, [Cevallos,2008].

Por esta razón, se dice que las redes de neuronas de base radial son redes con carácter local, ya que, dado un patrón de entrada a la red, solo aquellas neuronas ocultas cuyos centros estén en la vecindad de dicho patrón se van a activar; el resto de las neuronas ocultas permanecerán inactivas o con un menor nivel de activación, [Cevallos,2008].

Esto no sucede cuando se utiliza funciones de activaciones sigmoideas, como en el caso del perceptrón multicapa, pues estas se activan en todo un rango de valores. Esto hace que, aunque ambas arquitecturas sean redes multicapa con conexiones

hacia adelante, cada una de ellas posee sus propias características, así como sus ventajas e inconvenientes, [Cevallos, 2008].

2.2.5.3 *Aprendizaje de las redes de base radial*

El proceso de aprendizaje implica la determinación de todos los parámetros que intervienen en la red de base radial. Estos son: los centros y las desviaciones de las neuronas ocultas y los pesos de la capa oculta a la capa de salida, así como los umbrales de las neuronas de salida, [Cevallos,2008].

Debido a que las capas de neuronas en una red de base radial realizan tareas diferentes, es razonable separar el proceso de optimización de los parámetros de la capa oculta y los de la capa de salida mediante la utilización de diferentes técnicas. Así, para los parámetros de la capa oculta – centros y desviaciones - el proceso de aprendizaje debe estar guiado por una optimización en el espacio de patrones de entrada, pues cada una de las neuronas oculta de la red de base radial va a representar una zona diferente del espacio de entrada, sin embargo, para los parámetros de la capa de salida la optimización se debe realizar en base a la salida que se desea obtener o salidas deseadas, ya que las redes de base radial se utilizan para aproximar relaciones entre el conjunto de variables de entrada y salida que definen el problema. Por tanto, uno de los mecanismos más usados para el aprendizaje de las redes de base radial es el llamado método híbrido, que combina dos fases: Una fase no supervisada para la determinación de los centros y otra supervisada para la determinación de los pesos y umbrales, [Cevallos,2008].

2.2.5.4 *Método de aprendizaje híbrido*

El método híbrido realiza el aprendizaje de las redes de base radial en dos fases:

- **Fase no supervisada:** Puesto que las neuronas ocultas de las redes de base radial se caracterizan porque representan zonas diferentes del espacio de patrones de entrada, los centros y las desviaciones de las funciones de base radial deben ser determinados con este objetivo, es decir, con el objetivo de clasificar el espacio de entrada en diferentes clases. El representante de cada clase será el

centro de la función de base radial y la desviación vendrá dada por la amplitud de cada clase.

- **Determinación de los centros: algoritmos de K-medias**

Los centros de las funciones de base radial se determinan mediante un algoritmo de clasificación no supervisada que permita dividir el espacio de patrones de entradas en clases.

El número de clases es el número de neuronas ocultas en la red de base radial. El método más utilizado es el algoritmo de K-medias, aunque es necesario destacar que cualquier algoritmo de clasificación no supervisado podría ser utilizado, como, por ejemplo, los mapas auto-organizados Kohonen.

El algoritmo K-medias es un algoritmo de clasificación no supervisado mediante el cual el espacio de patrones de entrada se divide en k clases o regiones. El representante de cada una de estas clases C_i , será el centro de la neurona oculta i .

Dichos centros se determinan con el objetivo de minimizar las distancias euclidianas entre los patrones de entrada y el centro más cercano, es decir:

$$J = \sum_{i=1}^M \sum_{n=1}^N M_{in} \| X(n) - C_i \|^2 \quad (2.10)$$

Donde N es el número de patrones, $\| X(n) - C_i \|^2$ es la distancia euclidiana, $X(n)$ es el patrón de entrada n y M_{in} es la función de pertenencia, que vale 1 si el centro C_i es el más cercano al patrón $X(n)$, y cero en otro caso, es decir:

$$M_{in} = \begin{cases} 1 & \text{si } \|X(n) - C_i\| < \|X(n) - C_s\| \quad \forall s \neq i, s = 1, 2, \dots, k \\ 0 & \text{en otro caso} \end{cases} \quad (2.11)$$

Dado K el número de clases, $\{X(n) = (x_1(n), x_2(n), \dots, x_p(n))\} N=1 \dots N$ el conjunto de patrones de entrada y $\{C_i = (c_{i1}, c_{i2}, \dots, c_{ip})\} i=1 \dots k$ centros de las clases, los pasos para la aplicación del algoritmo son los siguientes:

Paso 1: Se inicializan los centros de las K clases. Pueden inicializarse a K patrones aleatorios del conjunto de patrones disponibles, o bien puede realizarse aleatoriamente, en cuyo caso es conveniente que se inicialicen dentro del rango de valores de los patrones de entrada.

Paso 2: Se asignan N_i patrones de entrada a cada clase i del siguiente modo: el patrón $X(n)$ pertenece a la clase i si $\|X(n)-C_1\| < \|X(n)-C_2\| \forall s \neq i, s=1, 2, \dots, K$.

Por tanto, cada clase tendrá asociado un determinado número de patrones de entrada, aquellos más cercanos al centro de la clase.

Paso 3: Se calcula la nueva posición de los centros de las clases como la media de todos los patrones que pertenecen a su clase, es decir:

$$c_{ij} = \frac{1}{N_i} \sum_{n=1}^m M_{in} x_j(n) \quad \text{Para } i=1,2,\dots,p \quad t=1,2,\dots,k \quad (2.12)$$

Paso 4: Se repiten los pasos 2 y 3 hasta que las nuevas posiciones de los centros no se modifiquen respecto a su posición anterior.

El algoritmo de K-medias es un método fácil de implementar y usar; suele ser un algoritmo bastante eficiente en problemas de clasificación, pues converge en pocas iteraciones hacia un mínimo de la función J dada por la ecuación 2.12, aunque podría tratarse de un mínimo local.

Uno de los inconvenientes o desventajas que se puede atribuir al algoritmo K-medias es su dependencia de los valores iniciales asignados a cada centro, lo cual hace que, en muchas ocasiones, y siempre dependiendo del problema, se obtengan soluciones locales.

▪ **Determinación de las amplitudes**

Una vez determinados los centros de las funciones de base radial, las amplitudes o desviaciones de dichas funciones deben calcularse de manera que cada neurona oculta se active en una región del espacio entrada y de manera que el solapamiento de las zonas de activación de una neurona a otra sea lo más ligero posible, para suavizar así la interpolación.

Las amplitudes de cada función de base radial se pueden determinar usando heurísticas, las cuales permiten que el solapamiento entre las neuronas ocultas sea lo más suave posible. Se puede utilizar diferentes heurísticas, como, por ejemplo:

- Media uniforme de las distancias euclidianas del centro C_i a los p centros más cercanos:

$$d_1 = \frac{1}{p} \sum_p \|C_1 - C_p\| \quad (2.13)$$

- Otra opción bastante eficiente es determinar la amplitud de la función de base radial como la media geométrica de la distancia del centro a sus dos vecinos más cercanos:

$$d_i = \sqrt{\|C_i - C_t\| \|C_i - C_s\|} \quad (2.14)$$

Siendo C_1 y C_2 y los dos centros más cercanos al centro C_i .

- **Fase supervisada**

En esta fase se calculan los pesos y umbrales de las neuronas de salida de la red. En este caso, el objetivo es minimizar las diferencias entre las salidas de la red y las salidas deseadas. Por tanto, el proceso de aprendizaje está guiado por la minimización de una función error computada en la salida de la red:

$$E = \frac{1}{N} \sum_{n=1}^N e(n) \quad (2.15)$$

Donde N es el número de patrones o muestras y $e(n)$ es el error cometido por la red para el patrón $X(n)$, que viene dado generalmente por:

$$e(n) = \frac{1}{2} \sum_{k=1}^r (s_k(n) - y_k(n))^2 \quad (2.16)$$

Siendo: $Y(n) = (y_1(n), \dots, y_r(n))$ y $S(n) = (s_1(n), \dots, s_r(n))$ los vectores de salida de la red y salida deseada para el patrón de entrada $X(n)$, respectivamente.

2.2.6 Redes de base radial frente a perceptrón multicapa

Las redes de base radial y el perceptrón multicapa son dos tipos de redes de neuronas no lineales que poseen sus neuronas agrupadas en capas y las conexiones están dirigidas hacia adelante –redes feedforward-. Ambas redes son aproximaciones universales, pues se ha demostrado que pueden aproximar cualquier función continua.

Estos dos tipos de redes presentan también algunas diferencias, como el número de capas ocultas –una única capa en el caso de las redes de base radial y tantas capas como se desee en el caso del perceptrón multicapa -en las redes de base radial las conexiones de la capa de entrada a la oculta no llevan pesos asociados- y, por ejemplo, también la linealidad en la capa de salida, la cual no es imprescindible para el caso del perceptrón multicapa.

Sin embargo, la principal diferencia entre las redes de base radial y el perceptrón multicapa radica en la función de activación de las neuronas ocultas de la red. Las primeras utilizan funciones de base radial, las cuales hacen que las neuronas ocultas de la red posean un carácter local, activándose cada neurona en una determinada región del espacio entrada. Por otro lado, el perceptrón multicapa emplea funciones de activación sigmoideas, que se activen en todo el espacio de entrada.

El uso de diferentes funciones de activación con las propiedades anteriormente mencionadas, hace que cada una de estas arquitecturas, perceptrón multicapa y redes de base radial, tenga sus propias características, las cuales se analizan a continuación:

- **El perceptrón multicapa construye aproximaciones globales:** debido al uso de funciones de activación sigmoidea, el perceptrón multicapa construye relaciones globales entre los datos de entrada y salida disponibles. Esto hace que el aprendizaje de la red sea lento, pues el cambio en un solo peso de la red provoca cambios en la salida para todos los patrones de entrada presentados anteriormente, reduciéndose así el efecto de previos ciclos de aprendizaje y retrasando la convergencia del algoritmo de aprendizaje.
- **Las redes de base radial construyen aproximaciones locales:** cada neurona oculta de la red de base radial se especializa en una determinada región del espacio de entrada y construyen una aproximación local de dicha región.

Por tanto, la relación que definen las redes de base radial entre los datos de entrada y salida es una suma de funciones no lineales y locales para diferentes regiones del espacio de entrada.

A diferencia de cuando se construyen aproximaciones globales, la construcción de aproximaciones locales permite que el aprendizaje sea más rápido, ya que el cambio en un solo peso de la red afecta únicamente a la neurona oculta asociada a dicho peso y, por tanto, a un determinado grupo de patrones de entrada, los pertenecientes a la clase que representa la neurona oculta en cuestión.

Debido al carácter local, el aprendizaje de estas redes es, generalmente, menos sensible al orden de presentación de los patrones que en el caso del perceptrón multicapa.

En muchos casos, sin embargo, ocurre que para poder construir una aproximación mediante la suma de aproximaciones locales se requiere un alto número de neuronas ocultas, lo cual podría influir negativamente en la capacidad de generalización de las redes de base radial.

Finalmente, debe señalarse que el número de neuronas ocultas de la red puede aumentar exponencialmente con la dimensión del espacio de entrada. Por tanto, para aplicaciones que requieren un alto número de variables de entrada, las redes de base radial podrían no ser las más adecuadas.

2.3 Teoría Sobre el Reconocimiento de Imágenes

2.3.1 Conceptos básicos de imágenes

Existen dos formas de representar imágenes digitalmente, formato vectorial y formato raster. El formato vectorial representa la imagen por sus formas internas, tratando de representar figuras geométricas que se acomodan a la imagen a representar. El formato raster en cambio representa la imagen a través de un conjunto rectangular (matriz o arreglo) de puntos coloreados llamados píxeles. Este formato es el más usado y de éste se hablará en lo que sigue.

2.3.2 Color

Cada pixel tiene un valor, este valor puede ser un simple valor o un conjunto de valores dependiendo de la cantidad de canales de color. En RGB, por ejemplo, cada pixel tendrá 3 valores de colores, uno para Red, otro para Green y otro para Blue que corresponde a su color.

La cantidad de bits (dígitos binarios 0 o 1) usados para representar dicho color se llama profundidad de color o colordepth, mientras más bits mayor calidad.

El color de un pixel puede ser representado de tres maneras:

- **Color indexado (indexed-color):** En este caso, el pixel posee un valor que en realidad no es un código de color, sino un índice del color real en una paleta de colores, también llamado mapa de colores o colormap. El formato GIF usa esta representación. Una paleta de colores no es más que una tabla indexada con los valores de colores a usar en la imagen. Esta paleta se incluye en la imagen. El colordepth en este caso indica la cantidad máxima de colores en la paleta, pero no la calidad de los colores de ésta.

- **Escala de grises (grayscale):** El valor del pixel en este caso es un tono de gris, donde cero indica negro y el mayor valor (2colordepth) indica blanco. La cual se utilizar para el reconocimiento de características de un rostro en el sistema propuesto.

- **Color verdadero (truecolor):** El valor del píxel es representado por un conjunto de tres valores Rojo, Verde, Azul (RGB).

2.3.3 Teoría de procesamiento de imágenes

2.3.3.1 *Conceptos generales*

La formación de una imagen digital es el primer paso para cualquier procesamiento de imágenes digitales, y consiste básicamente en un sistema óptico y el digitalizador,

mediante el cual la imagen óptica se transforma en una señal eléctrica que permitirá el procesamiento.

Al digitalizar una imagen, es común introducir ruido o degradación de la misma, por ello es importante considerar técnicas para restaurarla antes de procesarla; éstas consisten generalmente en disminuir el nivel de ruido, mejorar el contraste de la imagen, lograr la no-uniformidad de la imagen, su alineación, vecindad, etc.

Una vez digitalizada la imagen, puede considerarse como monocromática, susceptible de ser representada como una función $f_{(x,y)}$, donde x e y denotan coordenadas espaciales, y el valor de f (intensidad) en cualquier punto (x,y) es proporcional al nivel de gris (brillo) de la imagen en ese punto. Una imagen digital bidimensional es una imagen $f_{(x,y)}$ que ha sido separada en coordenadas espaciales y brillo (nivel de gris). Así mismo, una imagen digital puede ser convenientemente representada por una matriz I de tamaño $M \times N$ de la forma:

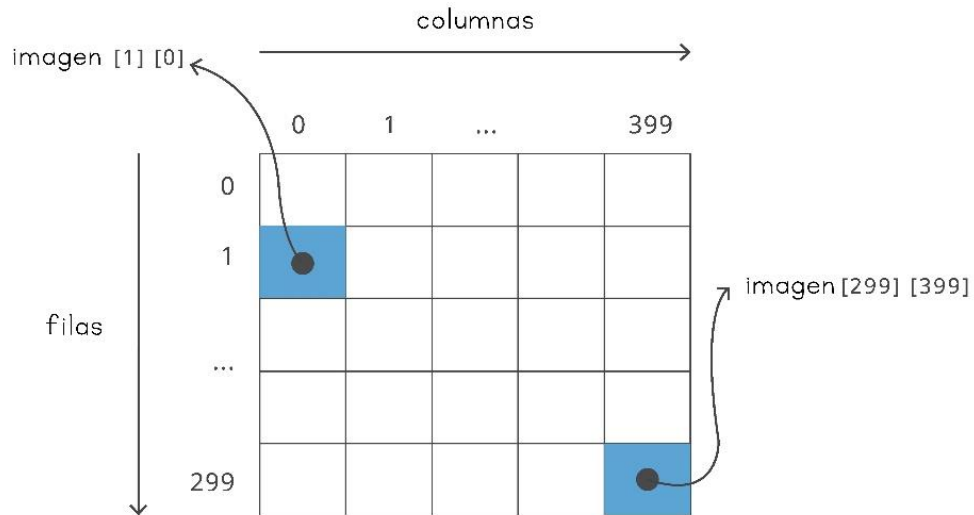
Figura 2.8 Coordenadas de una matriz.

$$I = \begin{matrix} I(1,1) & I(1,2) & \cdots & I(1,M) \\ I(2,1) & I(2,2) & \cdots & I(2,M) \\ \vdots & \vdots & & \vdots \\ I(N,1) & I(N,2) & \cdots & I(N,M) \end{matrix}$$

Fuente: [Elaboración propia, 2020].

En un arreglo de este tipo, los índices i, j de un sistema coordenado en una matriz corresponden a los ejes x e y respectivamente.

Figura 2.9 Definiciones básicas.



Fuente: [Elaboración propia, 2020].

2.3.3.2 Vecindad entre píxeles

Un píxel p en las coordenadas (x, y) tiene 4 vecinos horizontales y 4 verticales, cuyas coordenadas están dadas por:

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)$$

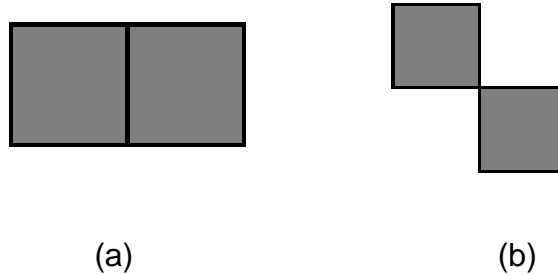
Este grupo de píxeles se nota como $N_4(p)$. Así mismo, las vecindades diagonales con el punto (x, y) se notan como $N_D(p)$, y sus coordenadas son:

$$(x + 1, y + 1), (x + 1, y - 1), (x - 1, y + 1), (x - 1, y - 1)$$

Para definir de forma adecuada el concepto de vecindad, es necesario revisar el de adyacencia. Dos píxeles son adyacentes si, y solo si, tienen en común una de sus fronteras, o al menos una de sus esquinas. La figura 1.2 muestra píxeles adyacentes.

Figura 2.10 Píxeles adyacentes.

(a) adyacentes por frontera. (b) adyacente por esquina.



Fuente: [Enrique C. 2004].

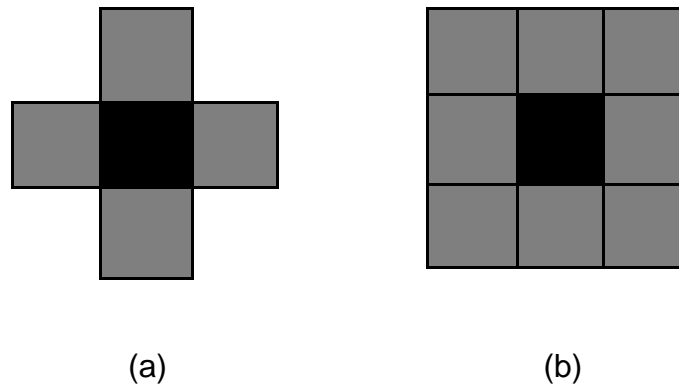
Dos píxeles son vecinos si cumplen con la definición de adyacencia. Si los píxeles comparten una de sus fronteras, se dice que los mismos son vecinos directos; si sólo se tocan en una de sus esquinas, se llaman vecinos indirectos.

Una vecindad de un píxel p_0 , denotada como V_p , es una submatriz $M \times L$ de tamaño $K \times L$, con K y L enteros impares pequeños, contenida en la matriz imagen (i_{MN}), la cual está formada por un número finito de píxeles vecinos o no de p_0 .

$$V_p = \{p: p \in MKL\}; MKL \subset i_{MN}; K = L = 3, 5, 9$$

Para ilustrar las definiciones anteriores, se puede observar, en la figura 1.3, vecindades de 4 y de vecindades de 8, la primera formada, por píxeles que son vecinos directos, mientras que la vecindad de 8 está formada tanto por vecinos directos como por indirectos.

Figura 2.11 Vecindades, (a) vecindad de 4, (b) vecindad de 8.



Fuente: [Enrique C. 2004].

En el desarrollo de técnicas de procesamiento de imágenes que involucren el análisis de una determinada región de la escena digital, es posible encontrar vecindades de 5x5 y hasta de 9x9; básicamente la definición de las dimensiones de la matriz vecindad depende de la técnica que se esté desarrollando.

2.3.3.3 Conectividad

La conectividad entre píxeles es un concepto utilizado para establecer los límites en objetos y regiones de componentes en una imagen.

Para establecer la conectividad entre dos píxeles, es necesario determinar si son adyacentes en sentido específico (si tiene 4 vecindades) y si su nivel de gris satisface un criterio especificado de similitud (si son iguales). Por ejemplo, en una imagen binaria con valores 0 y 1, dos píxeles pueden tener vecindad de 4, pero sólo se consideran conectados si tienen el mismo valor.

Considerando V como valores de niveles de gris utilizados para definir

Conectividad; se puede ejemplificar tomando una imagen binaria, $V = \{1\}$, para la conectividad de píxeles con valor 1. En una imagen de niveles de gris, para la conectividad de píxeles con valores en un rango de intensidad de, digamos, 32 a 64, sería:

$$V = \{2, 3, 4, \dots, 6, 7\}.$$

Existen tres tipos de conectividad.

- a) Conectividad 4. Dos pixeles, p y q, con valores de V, están conectados si q pertenece a $N_4(p)$.
- b) Conectividad 8. Dos pixeles, p y q, con valores de V, están conectados si q pertenece a $N_8(p)$.
- c) Conectividad mezclada. Dos pixeles, p y q, con valores de V, están conectados mezclados si:
 - ✓ q pertenece a $N_4(p)$, o bien
 - ✓ q pertenece a $N_D(p) \cap N_4(q)$ está vacío.

La conectividad mezclada es una modificación de la conectividad 8, y se introduce para eliminar las conexiones multi-trayectoria.

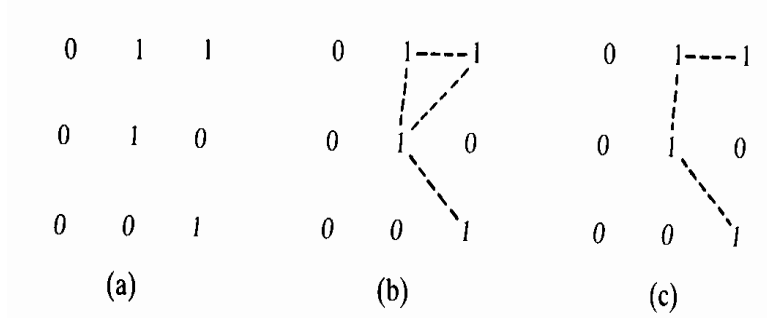
Un pixel p es adyacente a q si están conectados. Una trayectoria de pixel p con coordenadas (x, y) a un pixel q con coordenadas (s, t) es una secuencia de distintos pixeles con coordenadas:

$$(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n) \quad \text{donde } (x_0, y_0) = (x, y) \text{ y } (x_n, y_n) = (s, t)$$

Figura 2.12 Conectividad.

(a) Arreglo de pixeles; (b) Pixel central de conectividad 8;

(c) conectividad mezclada del mismo pixel.



Fuente: [Elaboración propia, 2020].

2.3.3.4 Relaciones y equivalencias.

Debemos tomar en cuenta algunos conceptos importantes que forman la base de las relaciones y equivalencias.

- Una relación binaria R en A es un grupo de pares de elementos de A . Si el par (a, b) pertenece a R , la notación utilizada generalmente es aRb , la cual se puede interpretar como “ a está relacionada con b ”. Una relación binaria R colocada en A se dice que es:
 - a) Reflexiva, si para cada punto a en A , aRa ;
 - b) Simétrica, si para cada a y b en A , aRb implica bRa ;
 - c) Transitiva, si para a , b y c en A , aRb y bRc implican aRc .

Una relación satisfactoria de estas tres propiedades se denomina relación de equivalencia. Si R es una relación equivalente en A , entonces A se puede dividir k veces; a esto se denomina clases de equivalencia. Para algunos k entre uno e infinito (255), incluso, algunos como aRb si y solo si a y b pertenecen a la misma clase de equivalencia.

2.3.3.5 Filtrado convolucional

Un filtro convolucional, aplicado a una imagen, se suele representar por un grupo de arreglos de filtros con una matriz de dimensión H renglones por W columnas. El filtro se ajusta para representar una frecuencia de filtrado específica cuando se aplica a la imagen.

El filtrado se aplica centrando la matriz $H \times W$ sobre cada pixel dentro de la imagen y realizando la convolución de los datos de la imagen con los del filtro:

Cuanto mayor sean los valores de H y W , el filtro convolucional tenderá con mayor precisión a la respuesta en la frecuencia deseada. H y W deben ser enteros impares

para que el filtro pueda centrarse en un pixel. Los valores de $H/2$ y $W/2$ se redondean hacia abajo al computarse la operación.

Un filtro pasa-altas ejemplificado en una matriz de 3×3 es:

Figura 2.13 Matriz 3×3 pasa-altas.

$$\begin{matrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{matrix}$$

Fuente: [Elaboración propia, 2020].

Un filtro pasa-bajas ejemplificado en una matriz de 3×3 es:

Figura 2.14 Matriz 3×3 pasa-bajas

$$\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$$

Fuente: [Elaboración propia, 2020].

La forma habitual de establecer el gradiente de una imagen $u(m, n)$ en un punto dado es mediante el producto de la imagen por dos máscaras, H_1 y H_2 , que representan la magnitud del gradiente en dos direcciones perpendiculares:

Un pixel se declara como perteneciente a un borde cuando $g(m, n)$ excede un determinado valor umbral T . En función de este valor T , se tendrá mayor o menor número de puntos de gradiente. Habitualmente se suele escoger este valor en función del histograma acumulado de $g(m, n)$, de forma que sólo del 5 al 10% de los puntos de máximo gradiente sea declarado como borde.

Cuadro 2.1 Muestra los operadores más comunes o máscaras que presentan las ventajas computacionales.

Figura 2.15 Operadores gradiente más comunes.

	Dirección horizontal	Dirección vertical
a) Roberts	$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$
b) Smoothed (o ewitt)	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
c) Sobel	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$
d) Isotrópico	$\begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$

Fuente: [Elaboración propia, 2020].

Se puede apreciar que los contornos extraídos con este método son dobles, y que siempre serán cerrados, aunque la región no lo sea.

2.3.3.6 Segmentación

La forma de un objeto puede ser descrita en términos de sus bordes o de la región que ocupa. La representación de objetos basada en los bordes requiere detectar contornos; para la representación basada en regiones, se realiza la segmentación de varias regiones homogéneas después de haber realizado la detección de contornos. La segmentación subdivide una imagen en partes u objetos; el nivel de subdivisión depende de la aplicación. La segmentación de imágenes vincula la división o separación de la imagen en regiones de atributos similares, considerando el atributo más básico de segmentación la amplitud, aunque los contornos y la textura de la imagen también han de ser considerados.

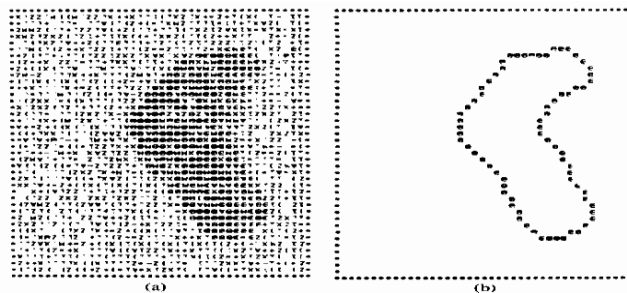
Los algoritmos de segmentación en imágenes monocromáticas suelen basarse en una o dos propiedades básicas de valores de nivel de gris: discontinuidad y similaridad.

2.3.3.7 *Thresholding*

Uno de los acercamientos a segmentación de imágenes más importantes es thresholding. Tradicionalmente, se ha efectuado una forma simple de definir el rango de valores de nivel de gris en la imagen original que consiste en elegir los pixeles en este rango según pertenezcan o no al fondo: se toman los que sí pertenecen y se rechazan todos los demás. Una imagen de este tipo se muestra como una imagen binaria (de dos niveles) utilizando blanco y negro u otros colores para distinguir las regiones (no hay una convención estándar sobre cuáles son los rasgos de interés, si los blancos o los negros, así que la elección varía en cada caso). Este tipo de operación se denomina thresholding, [Gutiérrez, 2002].

Suponiendo que el histograma de nivel de gris de la figura 1.4(a) corresponde a una imagen $f(x,y)$, compuesta por objetos brillantes sobre un fondo oscuro de tal forma que los pixeles de objetos y fondo son modos de selección, una forma obvia de extraer los objetos del fondo es seleccionar un umbral T que separe estos modos; después, cualquier punto (x,y) para el que $f(x,y) > T$ se denomina un punto del objeto; cualquier otro punto, se denomina punto del fondo, [Gutiérrez, 2002].

Figura 2.16 (a) Imagen ruidosa; (b) resultado de la detección de contornos.



Fuente: [Gutiérrez, 2002].

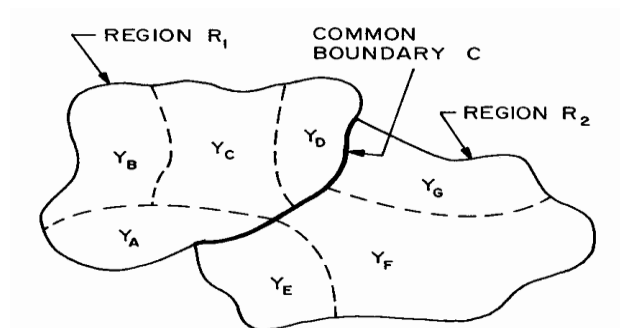
2.3.3.8 *Crecimiento de regiones*

El crecimiento de regiones es uno de los métodos conceptualmente más simples para la segmentación; los pixeles adyacentes de amplitud similar se agrupan juntos para formar una región segmentada. Sin embargo, en la práctica, hay limitantes,

algunas de las cuales son razonablemente simples y deben ser tomadas en cuenta para obtener un patrón de crecimiento que permita obtener resultados aceptables.

Brice y Fenema desarrollaron un método de crecimiento de regiones basado en un juego de reglas de crecimiento simples. En la primera parte del proceso, se combinan juntos pares de píxeles cuantizados en grupos denominados regiones atómicas si son de la misma amplitud, y regiones de conectividad 4. A continuación se aplican dos reglas heurísticas para disolver las vecindades débiles entre vecindades atómicas. Haciendo referencia a la figura 2.15., podemos observar dos regiones, R_1 y R_2 , adyacentes con perímetros P_1 y P_2 respectivamente, perímetros que han sido previamente unidos. Después de realizar las etapas iniciales del crecimiento de región, una región debe contener previamente sub-regiones unidas por diferentes valores de amplitud, [Tocona, 2007].

Figura 2.17 Geometría de crecimiento de regiones.



Fuente: [Tocona, 2007].

Considerando C como el tamaño de la vecindad común y Y el tamaño de la porción de C en que la diferencia de amplitud Y a través de la región es menor al factor de significancia ϵ , las regiones R_1 y R_2 están unidas si:

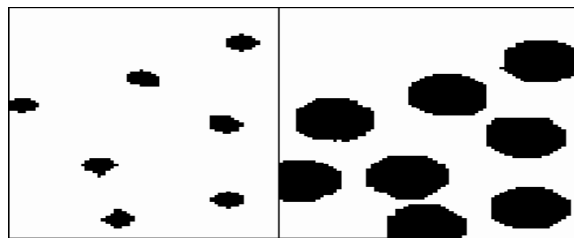
El método de crecimiento de Brice y Fenema ofrece buen rendimiento para segmentación de imágenes simples y de poca textura.

2.3.3.9 Dilatar

La operación de dilatar una imagen se puede describir como un crecimiento de píxeles, en otras palabras, se marca con 1 parte del fondo de la imagen que toque un píxel que forma parte de la región. Esto permite aumentar un nivel de píxeles alrededor de la periferia de cada región y así incrementarlo en dimensión y rellenar hoyos dentro de la región.

Por tanto, el proceso de dilatación consiste en obtener la reflexión de B sobre su original y luego trasladar esta reflexión por x. La dilatación de A y B entonces es el grupo de todas las x desplazadas cuando x y A se traslapan por al menos un elemento diferente de cero. Dado que la dilatación aumenta el tamaño de una región, algunas veces se conoce como crecimiento, [Tocona, 2007].

Figura 2.18 Ejemplo de imagen dilatada.



Fuente: [Tocona, 2007].

2.3.3.10 Cerrar

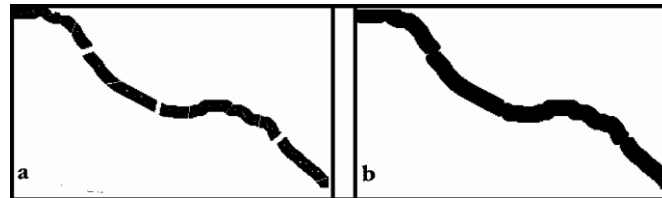
El proceso de dilatar y a continuación erosionar (proceso inverso), se conoce como de cerrado, en otras palabras, el proceso de cerrar morfológicamente una imagen consiste en marcar con un 1 los píxeles aislados que se encuentran entre ceros. Es decir, suaviza secciones del contorno, pero generalmente une separaciones estrechas y golfos delgados; elimina pequeños hoyos y rellena aberturas.

Cerrar un grupo A estructurando el elemento B se define como: $A \bullet B = (A \oplus B) \ominus B$

Es decir, para cerrar A por medio de B, se dilata A y se erosiona el resultado de B.

Estas propiedades, aunadas a los resultados de las operaciones de cerrado, se utilizan para construir filtros morfológicos, [Eslava, 2012].

Figura 2.19 Ejemplo de cerrar una imagen.



Fuente: [Eslava, 2012].

2.3.4 Reconocimiento de patrones de una imagen

El reconocimiento de patrones se utiliza actualmente para la solución de tareas tales como el reconocimiento de caracteres, de huellas digitales y reconocimiento de voz, entre muchos otros. Por esta razón, desde hace varios años se han venido desarrollando diferentes metodologías que intentan resolver este tipo de problemas.

¿Qué es un patrón?: Siguiendo la definición de Watanabe, un patrón es una entidad a la que se le puede dar un nombre y que está representada por un conjunto de propiedades medidas y las relaciones entre ellas (vector de características). Por ejemplo, un patrón puede ser una señal sonora y su vector de características el conjunto de coeficientes espectrales extraídos de ella (espectrograma). Otro ejemplo podría ser la imagen de una cara humana de las cuales se extrae el vector de características, formado por un conjunto de valores numéricos calculados a partir de esta imagen.

El reconocimiento automático, descripción, clasificación y agrupamiento de patrones son actividades importantes en una gran variedad de disciplinas científicas, como: biología, psicología, medicina, visión por computador, inteligencia artificial, teledetección, entre otras.

Un sistema de reconocimiento de patrones tiene uno de los siguientes objetivos:

- Identificar el patrón como miembro de una clase ya definida (clasificación supervisada).
- Asignar el patrón a una clase todavía no definida (clasificación no supervisada, agrupamiento o clustering).

El diseño de un sistema de reconocimiento de patrones se lleva a cabo normalmente en tres fases:

- Adquisición y pre proceso de datos.
- Extracción de características.
- Toma de decisiones o agrupamiento.

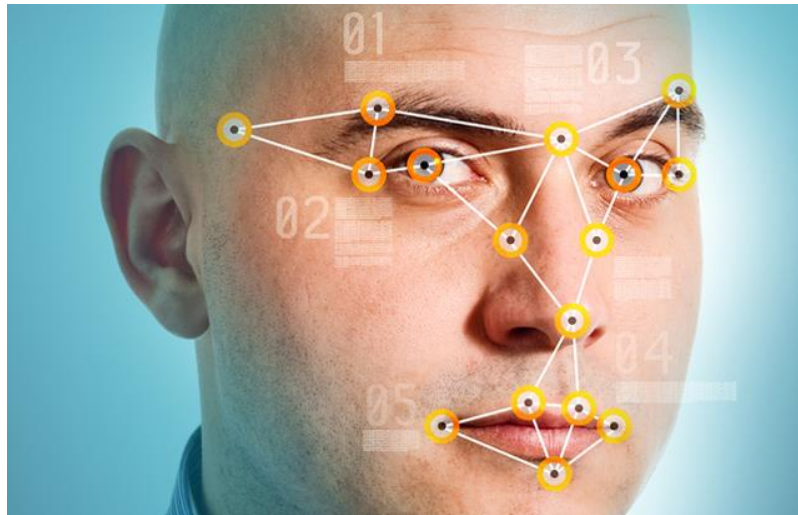
El universo del discurso, o dominio del problema, gobierna la elección de las diferentes alternativas en cada paso: tipo de sensores, técnicas de pre procesamiento, modelo de toma de decisiones, etc. Este conocimiento específico del problema está implícito en el diseño y no se representa como un módulo separado como sucede, por ejemplo, en los sistemas expertos. Tradicionalmente, el reconocimiento de patrones se ha abordado desde un punto de vista estadístico, dando lugar al llamado Reconocimiento Estadístico de Patrones (REP), del cual se dan unas breves pinceladas en el punto siguiente. No obstante, existe una alternativa que se ha revelado como muy prometedora en algunos casos en que el REP no funciona satisfactoriamente. Dicha alternativa son las Redes Neuronales Artificiales (RNA) que se analizarán más adelante, por ser el fundamento de este trabajo. Finalmente, se presenta una visión de los puntos comunes entre ambas técnicas.

2.3.4.1 *Etapas para el reconocimiento de patrones de una imagen*

De manera general, cuando se intenta resolver algún problema que implique el reconocimiento de patrones, se debe de tomar en cuenta las siguientes etapas:

- Adquisición de datos.
- Pre procesamiento.
- Extracción de características.
- Reconocimiento.

Figura 2.20 Patrones de reconocimiento.



Fuente: [Paz, J, 2009].

a) Adquisición de Datos

Para poder realizar el reconocimiento es necesario, realizar o implementar alguna etapa de adquisición de los datos que describen al patrón que se desea clasificar. Esto implica que, el método que se utilizará para realizar la adquisición de los datos, dependerá de la naturaleza del patrón. Por ejemplo, si se desea desarrollar un sistema para reconocer caracteres, esta etapa se podría realizar con la ayuda de una cámara digital o utilizando un escáner. Fuente, [Paz, J,2009].

b) Pre procesamiento

Una vez que ya se cuenta con los datos que describen a cada uno de los patrones, es más conveniente realizar una etapa de pre procesamiento sobre cada uno de ellos, en lugar de ser dados como entrada del sistema tal y como fueron obtenidos durante la etapa de adquisición de datos.

El realizar un pre procesamiento sobre los datos tiene varias ventajas; la principal de ellas es que puede reducir la dimensión de los datos, lo cual mejora substancialmente la ejecución del sistema, sobre todo cuando se utiliza una metodología como la de redes neuronales.

Dentro de esta etapa se puede pre procesar el patrón de entrada de tal forma que todos los patrones tengan el mismo tamaño (escala), consiguiendo con esto que el sistema sea invariante al escalamiento. Además de esto, también se busca lograr que el sistema sea invariante a la traslación. Cuando un sistema es invariante a la traslación y al escalamiento de los patrones, se dice que el sistema cuenta con un conocimiento previo.

c) Extracción de Características

Uno de los principales problemas en el reconocimiento de patrones, es encontrar una manera óptima de representar la información original que describe a cada uno de los patrones. Este problema es conocido como extracción de características. Este proceso de extracción de características trata de reducir la cantidad de información (reducción de dimensión) 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.

La extracción de características debe de cumplir con las siguientes condiciones:

- La dimensión del vector de características debe de ser menor que la del patrón original.
- Las características deben representar una codificación óptima de la entrada, perdiendo la información que no sea muy importante.

Especialmente, cuando se realiza el reconocimiento de patrones utilizando redes neuronales, la extracción de características debe tratar de obtener un vector de características con una dimensionalidad mucho menor a la del patrón original, puesto que, un vector con una dimensionalidad más pequeña que sea dado como entrada a la red neuronal tiene varios beneficios. En primer lugar, la cantidad de pesos que deben de ser aprendidos es menor, y, en segundo lugar, al tener menos pesos, el tiempo de entrenamiento puede ser reducido considerablemente, Fuente: [Paz, J, 2009].

d) Reconocimiento

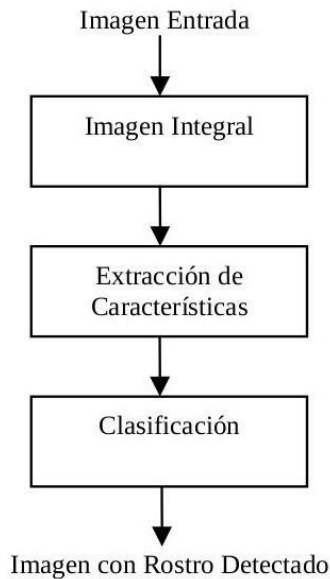
Finalmente, una vez que se han realizado las etapas de pre procesamiento y extracción de características, se procede a realizar el objetivo principal del sistema: el reconocimiento. De manera general, la etapa de reconocimiento consiste en obtener una salida del sistema como respuesta a un conjunto de estímulos dados como entrada (patrones).

2.4 Teoría de Viola – Jones

El método de detección de objetos de Viola Jones fue propuesto por Paul Viola y Michael Jones en 2001, que fue el primer método que proporciona tasas de detección de objetos relativamente altas. Se puede utilizar para detectar objetos en tiempo real, pero se aplica principalmente a la detección de rostros. La tasa de detección de este método es relativamente alta y muy baja con relación a falsos positivos, lo que hace al algoritmo tan robusto y procesa las imágenes rápidamente. Su objetivo principal es la detección de rostros mas no el reconocimiento, es decir, distinguir las caras de las no caras, este procedimiento es el primer paso para el reconocimiento Facial.

La metodología se basa en la propuesta. Esta se divide en tres etapas como se muestra en la Figura 2.18, en la primera etapa se realiza una transformación de la imagen generando una nueva llamada imagen integral, en el segundo bloque se realiza la extracción de características usando filtros con base haar¹, y por último se usa boosting² para la construcción de clasificadores en cascada, [hernandez, 2010].

Figura 2.21 Diagrama de bloques.



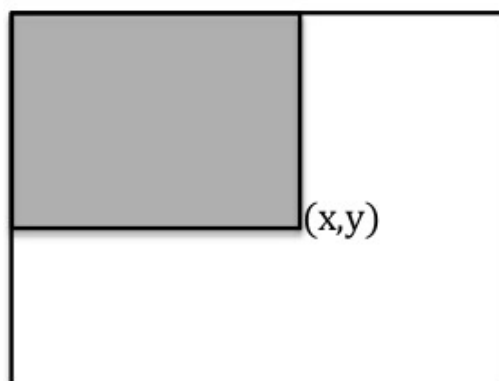
Fuente: [hernandez, 2010].

2.4.1 Características

2.4.1.1 *Imagen integral*

Esta nueva representación de una imagen fue introducida. Esta imagen permite extraer de forma rápida características a diferentes escalas ya que no se trabaja directamente con los valores de intensidad si no con una imagen acumulativa que se construye a partir de operaciones básicas.

Figura 2.22 Imagen integral.



Fuente: [hernandez, 2010].

La imagen integral (Figura 2.19) en la localización x, y , contiene la suma de los pixeles de la parte superior izquierda de la imagen. La suma de todos los pixeles de cualquier rectángulo en la imagen como se puede calcular a continuación

$$II(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y')$$

donde $II(x, y)$ es la imagen integral e $I(x, y)$ es la imagen original.

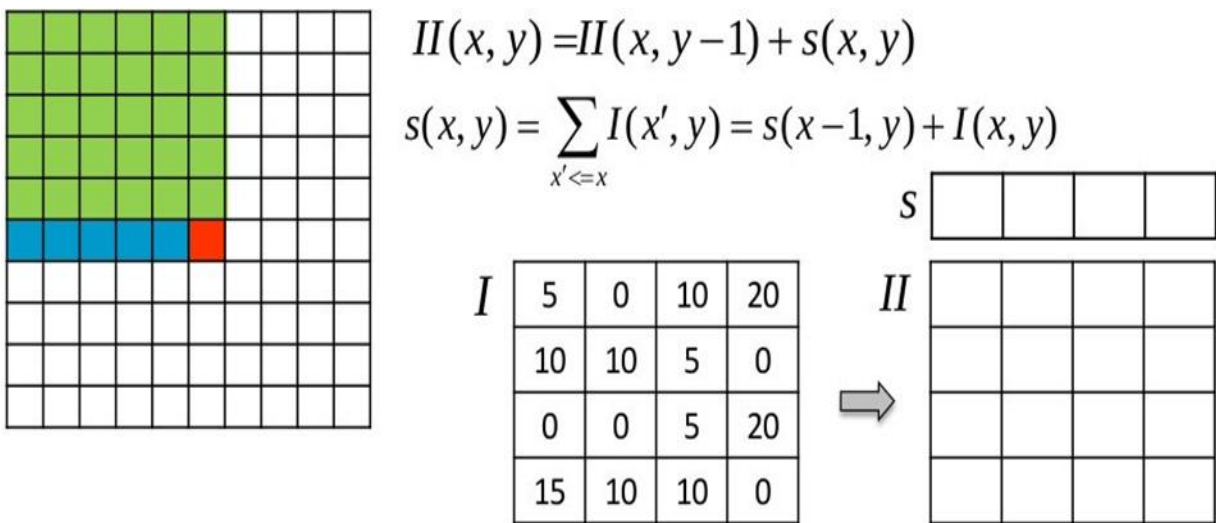
2.4.1.2 Cálculo de imagen integral

Un cálculo muy eficiente con sólo un recorrido por toda la imagen a partir de la suma acumulada de la fila actual y el valor de la imagen integral en la fila anterior.

2.4.1.3 Características Haar

En imágenes las características de cada objeto se extraen al aplicar ciertas funciones que permitan la representación y descripción de los objetos de interés de la imagen (patrones). La extracción de características es un paso en el reconocimiento de patrones en el cual las medidas u observaciones son procesadas para encontrar atributos que puedan ser usados para asignar.

Figura 2.23 Cálculo de imagen integral.



Fuente: [hernandez, 2010].

objetos a determinada clase.

En la metodología seguida, la extracción de características es realizada aplicando a la imagen filtros con bases haar. Estos filtros pueden ser calculados eficientemente sobre la imagen integral, son selectivos en la orientación espacial y frecuencia, y permiten ser modificados en escala y orientación. En la Figura 3.4 se muestran algunos de los filtros usados para la extracción de características.

De esta manera, unos conjuntos de características pueden ser usadas para codificar los contrastes encontrados en los rostros.

La propuesta original considera 3 tipos características:

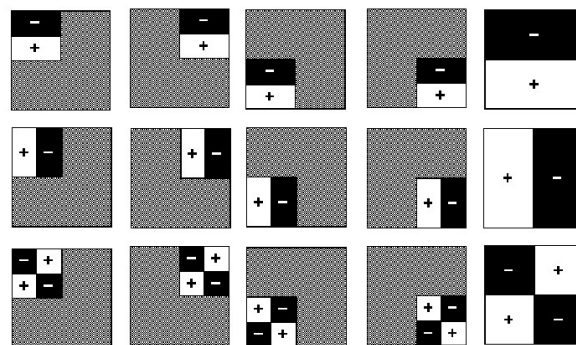
Característica de dos rectángulos: Su valor se calcula con la diferencia entre la suma de los pixeles dentro de las dos regiones rectangulares.

Característica de tres rectángulos: Su valor se calcula con la suma de los pixeles dentro de las dos regiones rectangulares exteriores y la sustracción de los pixeles de la región interior

Característica de cuatro rectángulos: Su valor se calcula con la diferencia entre las diagonales de los pares de rectángulos.

Rectángulos en negro representan zonas con una contribución positiva al filtro. Rectángulos en blanco representan zonas con una contribución negativa al filtro (hernandez, 2010).

Figura 2.24 Filtros Haar rotados, trasladados y con cambios de escala.



Fuente: [hernandez, 2010].

Figura 2.25 Representación de filtros Haar.

200	200	100	100	200	200	100	100
250	250	50	50	250	250	50	50
255	255	255	255	100	100	100	100
255	255	255	255	100	100	100	100
200	200	100	100	200	200	100	100
250	250	50	50	250	250	50	50
255	255	255	255	100	100	200	200
255	255	255	255	100	100	250	250

Fuente: [hernandez, 2010].

2.5 Teoría de la Biometría

Figura 2.26 Cascada de clasificadores.



Fuente: [hernandez, 2010].

La biometría se basa en la premisa de que cada individuo es único y posee rasgos físicos distintivos (rostro, huellas digitales, iris de los ojos, etc) o de comportamientos (la voz, la manera de firmar, etc), los cuales pueden ser utilizados para identificarla o validarla. Los dispositivos capaces de realizar el proceso de identificación o validación son el tema de este estudio.

Desde sus primeras apariciones en el mercado, este tipo de dispositivos han tenido que sortear tres dificultades fundamentales:

- Su elevado coste que impedía su despliegue masivo, de cientos o incluso de miles de unidades en las grandes corporaciones donde, cada empleado, debería poseer sus propios dispositivos de seguridad.

- Su tamaño, demasiado grande para poder instalarlo normalmente en ordenadores de sobremesa, portátiles o en los pequeños dispositivos de mano tan de moda en la actualidad como teléfonos móviles, PDA's, etc.
- Y por último, la poca sensibilidad mostrada por los grandes suministradores de redes hacia el uso y la necesidad de integración de este tipo de productos biométricos en sus infraestructuras de red.

Este panorama está cambiando drásticamente en estos últimos meses como consecuencia del interés y de la necesidad creciente surgida en el mercado a la hora de exigir sistemas más seguros. (Mercado internacional). La medición biométrica ha venido estudiándose desde tiempo atrás y es considerada en la actualidad el método ideal de identificación humana (hernandez, 2010).

2.5.1 Biometría

La palabra biometría deriva de las palabras: bio (vida) y metría (medida). La ciencia biométrica se define como el análisis estadístico de observaciones biológicas.

Así, un dispositivo biométrico es aquel que es capaz de capturar características biológicas de un individuo (rostro, huella dactilar, voz, etc.), compararlas, electrónicamente, contra una población de una o más de tales características y actuar según el resultado de la comparación (Saavedra, 2015).

2.5.2 Técnicas de autenticación

La autenticación de usuario es básicamente un proceso de prueba de su identidad. Existen tres técnicas básicas, basada en: "algo que el usuario sabe", "algo que el usuario tiene" y "algo que el usuario es". Las señas de acceso corresponden a la categoría "algo que el usuario sabe" por ejemplo, PASSWORD. Las tarjetas de crédito y llaves comunes son ejemplos de técnicas "algo que el usuario tiene". La huella digital, la voz, la retina, el rostro, son ejemplos.

La autenticación "algo que el usuario sabe": asume que solo una persona conoce una determinada clase o seña de acceso. Este tipo de autenticación se usa

ampliamente en redes, Internet y en las intranets. Una gran amenaza que trae esta técnica es que un impostor descubra la contraseña de alguien y se haga pasar por esta persona, violando la seguridad. Existen formas de disminuir la posibilidad de que alguien descubra la seña, por ejemplo, aumentando la longitud de la contraseña. El problema de aumentar el tamaño de la clave radica en que los seres humanos no somos buenos para recordar códigos largos, lo cual hace que, aunque el sistema admita contraseñas largas, pocas las utilicen.

La técnica “algo que el usuario tiene”: supone que el usuario legítimo es una persona que tiene un dispositivo determinado (tarjeta, llave, control remoto, etc) que lo autentica ante el sistema. Si alguien roba el dispositivo, o él legítima usuario lo pierde, el sistema no vale nada. Por ejemplo, un automóvil, quien tenga las llaves, es el “dueño del auto”.

La técnica “algo que el usuario es” n intenta basarse en alguna característica biológica y física del usuario. Puede autenticar, por ejemplo: las huellas dactilares, el iris, el rostro humano, la palma de la mano, DNA, etc.

Existe una variedad de dispositivos biométricos todos son bastante caros para usarse en todos los casos donde se requiera seguridad. Sin embargo, existen situaciones en las cuales el valor de lo que se quiere resguardar justifica el costo del sistema.

Los dispositivos más usados actualmente están basados en el reconocimiento del iris, las huellas digitales, el reconocimiento facial, la caligrafía y de patrón de voz (Saavedra, 2015).

2.5.2.1 Combinando técnicas

A fin de aumentar la seguridad, muchos sistemas combinan las técnicas anteriores. Por ejemplo, los cajeros automáticos. Aquí el usuario debe, primero introducir su tarjeta, “algo que el usuario tiene”, luego de que el sistema la reconoce como válida, debe ingresar su PIN “algo que el usuario sabe”.

Es común, también, que, al comprar con tarjeta de crédito, la vendedora solicite la cédula de identidad policial. Aquí el sistema combina las técnicas “algo que el usuario tiene” y “algo que el usuario es” usando de esta forma una técnica biométrica primitiva.

Se utiliza también un sistema en el cual un individuo debe primero ingresar un código alfanumérico, y luego presentar al sistema alguna característica biométrica, por ejemplo, la palma de la mano. En este sistema se combinan las técnicas, “algo que el usuario es” con “algo que el usuario sabe” (Saavedra, 2015).

2.5.3 Usos de Biometría

La biometría es fácil de usar, nada que recordar nada que cambiar nada que perder. Además, proporciona un nivel más alto de seguridad, unívoca “firma” de una característica humana que no puede ser fácilmente adivinada o “hackeada”. La Identificación y Autenticación biométrica explota el hecho de que ciertas características biológicas son singulares e inalterables y son, además, imposibles de perder, transferir u olvidar. Esto las hace más confiables, amigables y seguras que los password.

En el pasado el procesamiento biométrico era hecho manualmente por gente que física y mentalmente comparaba huellas dactilares contra tarjetas, rostros contra fotos de pasaportes y voces contra cintas grabadas. Hoy en día, dispositivos tales como escáneres, videocámaras, y micrófonos pueden, electrónicamente, capturar y entregar estas mismas características biométricas para automatizar procesos y comparaciones. Cada tecnología biométrica (huella dactilar, rostro, voz, etc) tiene sus propias características, variedades y certezas.

El proceso de captura, extracción de esas características y variedades, el almacenamiento y la comparación es universalmente similar para todos los dispositivos biométricos. Pero no todo es perfecto en estos sistemas. Existe la posibilidad de que el sistema acepte o rechace indebidamente a un usuario. Existen algoritmos que permiten minimizar estos errores.

Los niveles de precisión biométricos pueden variar, pero son siempre más confiables que el 100% de falsas aceptaciones experimentadas con las passwords prestadas o robadas. (Saavedra, 2015).

2.5.4 Funcionamiento básico de dispositivos biométricos

Figura 2.27 Sistemas biométricos.



Fuente: [Saavedra, 2015].

La mayoría de los sistemas biométricos funcionan de maneras muy similares y se puede resumir en dos pasos:

El primer paso consiste en que la persona debe registrarse (“enroll” en inglés) en el sistema. Durante el proceso de registro, el sistema captura el rasgo característico de la persona, como por ejemplo el rostro de la persona, y lo procesa para crear una representación electrónica llamada modelo. El modelo de referencia debe ser guardado en una base de datos, una tarjeta inteligente o en algún otro lugar del cual será extraído en cualquier ocasión futura para el segundo paso. A pesar de que es poco probable obtener dos tomas iguales aún del mismo individuo, a causa de diferencias ambientales y otras condiciones en el momento de la captura, el sistema aún debe poder funcionar correctamente. La mayoría de los algoritmos de comparación generan un ámbito para cada ensayo de comparación el cual es cotejado dentro de determinados umbrales antes de ser aceptados o rechazados. Cada

proveedor de tecnología biométrica configura la/el falsa/o aceptación/rechazo de forma diferente.

Las tasas de errores son medidas de dos maneras, una por la cantidad de personas con permiso que son rechazadas (tasa de falso rechazo) y otro por la cantidad de personas sin permiso que son aceptadas (tasa de aceptación indebida). En este caso, es claro, que la mayor preocupación se centra con el segundo tipo, pero en implementaciones prácticas el primer problema genera mucha molestia.

Si el umbral es demasiado bajo, se vuelve demasiado fácil para una persona no autorizada ser aceptada por el sistema, en cambio si el umbral está demasiado alto, personas autorizadas pueden llegar a ser rechazadas.

De acuerdo a la teoría tradicional en biometría, el segundo paso depende de si la función del sistema biométrico consiste en verificar la identidad de la persona o identificar a la persona.

En el caso de verificación, la persona le informa al sistema cuál es su identidad ya sea presentando una tarjeta de identificación o entrando alguna clave especial. El sistema captura el rasgo característico de la persona (la huella digital en nuestro ejemplo) y lo procesa para crear una representación electrónica llamada modelo en vivo. Por último, el sistema compara el modelo en vivo con el modelo de referencia de la persona. Si ambos modelos parecen la verificación es exitosa. De no serlos la verificación es fallida.

En caso de que la función del sistema biométrico sea identificación, la persona no le informa al sistema biométrico cuál es su identidad. El sistema tan solo captura el rasgo característico de la persona y lo procesa para crear el modelo en vivo. Luego el sistema procede a comparar el modelo en vivo con un conjunto de modelos de referencia para determinar la identidad de la persona.

Dependiendo de la función del sistema, este segundo paso puede ser:

- **Identificación positiva**

La función de un sistema de identificación positiva consiste en probar que la identidad de la persona está registrada en el sistema. La persona hace una reclamación positiva de identidad al sistema biométrico, es decir, la persona alega que está registrada en el sistema. El sistema responde comparando automáticamente el modelo en vivo con uno o varios modelos de referencia. Si la persona es identificada, el sistema biométrico le concede a la persona ciertos privilegios, de lo contrario los privilegios son negados.

- **Identificación negativa**

La función de un sistema biométrico de identificación negativa consiste en probar que la identidad de la persona no está registrada en el sistema biométrico. Un ejemplo puede ser un sistema que verifique que las personas que entran a un banco no se encuentren en una lista de delincuentes. La persona le hace una reclamación negativa de identidad al sistema biométrico, el cual responde comparando automáticamente el modelo en vivo con uno o varios modelos de referencia. Si la identidad no está registrada, el sistema biométrico le concede ciertos privilegios a la persona como, por ejemplo, permitirle entrar al banco. Si el sistema reconoce a la persona, este le niega dichos privilegios y hasta quizás alerte si se debe tomar alguna acción más radical como intervenir la persona.

2.6 Teoría Sobre el Desarrollo del Prototipo

El desarrollo del prototipo empieza con el registro de los docentes de la carrera Ingeniería Autotróica, empezando por los datos generales y tomando 10 fotografías de cada uno, de una posición frontal y directa a la cámara web y una distancia entre 40 cm a 50 cm, distancia que será decisiva al momento del reconocimiento del rostro ingresado al sistema de la cámara web.

En forma general se puede decir que este proceso va desde la transformación de la imagen ingresada al sistema, a escala de grises, para primero identificar un rostro en la imagen y posterior extracción de los puntos característicos, las cuales son los ojos,

nariz y boca, formando con esos puntos unas rectas que nos dan distancias entre ellos.

Los mismos que serán usados para entrenar nuestra red neuronal artificial de Función Base Radial, posterior a esto se realizan varios procesos con cada una de las imágenes tomadas y que ingresaran al prototipo del sistema, para llegar al objetivo que es el reconocimiento de las personas, que se encuentran registradas en el sistema.

2.7 Métricas de calidad de software

El Principal objetivo de los ingenieros de software es producir sistemas, aplicaciones o productos de alta calidad. Para las evaluaciones que se quieran obtener es necesario la utilización de medidas técnicas, que evalúan la calidad de manera objetiva (Pereira B, 2010).

La métrica de calidad de software que se utilizará en el trabajo de investigación, son las Normas ISO 9000 ISO/IEC 9126, con atributos detallados en la

Figura 2.28 Características de la norma ISO/IEC 9126.



Fuente: [Pereira B, 2010].

2.8 Modelo de Costo Cocomo

2.8.1 Método de estimación de costo COCOMO II

COCOMO II permite realizar estimaciones en función del tamaño del software y de un conjunto de factores de coste y de escala.

- En los factores de coste se incluyen aspectos relacionados con la naturaleza del sistema, equipo y características propias de proyecto.
- Los factores de escala incluyen la parte de escala producida a medida que un proyecto de software incrementa su tamaño.

Por un lado, COCOMO II define tres modos de desarrollo de software o tipos de proyectos las cuales son las siguientes.

2.8.1.1 *Orgánico*

Proyectos relativamente sencillos, menores de 50 KLDC líneas de código, en los cuales se tiene experiencia de proyectos, se encuentra en entornos estables.

2.8.1.2 *Semi - acoplado*

Proyectos intermedios en complejidad y tamaño, menores a 300 KLDC líneas de código, donde la experiencia en este tipo de proyectos es variable, y las restricciones intermedias.

2.8.1.3 *Empotrado*

Proyectos bastante complejos, en los que apenas se tiene experiencia y se engloba en un entorno de gran innovación técnica, además se trabaja con unos requisitos muy restrictivos y de gran volatilidad.

Tabla 2.1 Detalle de coeficientes de COCOMO II.

MODO	Básico				Intermedio			
	a	B	c	D	A	B	C	D
Orgánico	2,4	1,05	2,5	0,38	3,2	1,05	2,5	0,38
Semi acoplado –	3,0	1,12	2,5	0,35	3,0	1,12	2,5	0,35
Empotrado	3,6	1,20	2,5	0,32	2,8	1,20	2,5	0,32

Fuente: [Roger Pressman, 2010].

Por otro lado, existen diferentes modelos que define COCOMO II y son las siguientes:

2.8.1.4 Modelo básico

Se basa exclusivamente en el tamaño expresado en LDC.

2.8.1.5 Modelo intermedio

Además del tamaño del programa incluye un conjunto de medidas subjetivas llamadas conductores de costes.

2.8.1.6 Modelo avanzado

Incluye todo lo del modelo intermedio además del impacto de cada conductor de coste en las distintas fases de desarrollo.

Para la realización del COCOMO previamente necesitamos conocer el número de líneas de código, posteriormente para poder realizar los cálculos del método de estimación usamos las siguientes ecuaciones.

Tabla 2.2 Ecuaciones del Método COCOMO II.

Variable	Ecuación	Tipo/Unidad
Esfuerzo	$E = a (KLCD)^b$	Personas/mes
Tiempo	$T = c (E)^d$	Meses
Número de personas	$NP = \frac{E}{T}$	Personas
Costo total	CT = NP x T x sueldo mes	\$us.
Costo por líneas de código	$Costo LDC = \frac{CT}{LDC}$	\$us.

Fuente: [Calibrado, 1981].

2.8.1.7 Descripción de Valores Atributos de software

RELY: garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto.

DATA: tamaño de la base de datos en relación con el tamaño del programa.

CPLX: representa la complejidad del producto.

2.8.1.8 Atributos de hardware

TIME: limitaciones en el porcentaje del uso de la CPU.

STOR: limitaciones en el porcentaje del uso de la memoria.

VIRT: volatilidad de la máquina virtual.

TURN: tiempo de respuesta requerido.

2.8.1.9 Atributos del personal

ACAP: calificación de los analistas.

AEXP: experiencia del personal en aplicaciones similares.

PCAP: calificación de los programadores.

VEXP: experiencia del personal en la máquina virtual.

LEXP: experiencia en el lenguaje de programación a usar.

2.8.1.10 Atributos del proyecto

MODP: uso de prácticas modernas de programación.

TOOL: uso de herramientas de desarrollo de software.

SCED: limitaciones en el cumplimiento de la planificación. Disponible en: (COCOMO, 2016).

2.9 Seguridad de la información

2.9.1 Estándar ISO/IEC 27000.

Es un estándar para la seguridad de la información fue aprobado y publicado como estándar internacional en octubre del 2005 por ISO. Especifica los requisitos necesarios para establecer, implantar, mantener y mejorar un Sistema de Gestión de la Seguridad de la Información (SGSI).

SGSI (sistema de gestión de la seguridad de la información), es el conjunto de políticas y procedimientos que normalizan la gestión de la seguridad de la información, de toda una organización o de uno o varios procesos de negocio, debe garantizar la disponibilidad, confidencialidad e integridad de la información.

3. MARCO APLICATIVO

3.1 Diagnóstico Actual de Ingeniería Autotrónica

La carrera de Ingeniería Autotrónica no cuenta con un sistema biométrico de control de personal, el control tanto del personal Docente, se los realiza de manera manual, escrita en partes diarios para posteriormente, presentadas en Decanatura del Área “Desarrollo Tecnológico Productivo” y enviadas a Recursos Humanos de la Universidad Pública de El Alto y una copia del mismo son archivadas por Ingeniería Autotrónica.

Las planillas son registradas diariamente y son usadas como una base de datos que sirve como respaldo de la asistencia de clases, y son usados para extraer de ellos, los datos que se requiera, para dar los informes diarios, semanales, mensuales, anuales y los requerimientos de Ingeniería Autotrónica.

3.2 Descripción de Ingeniería Autotrónica

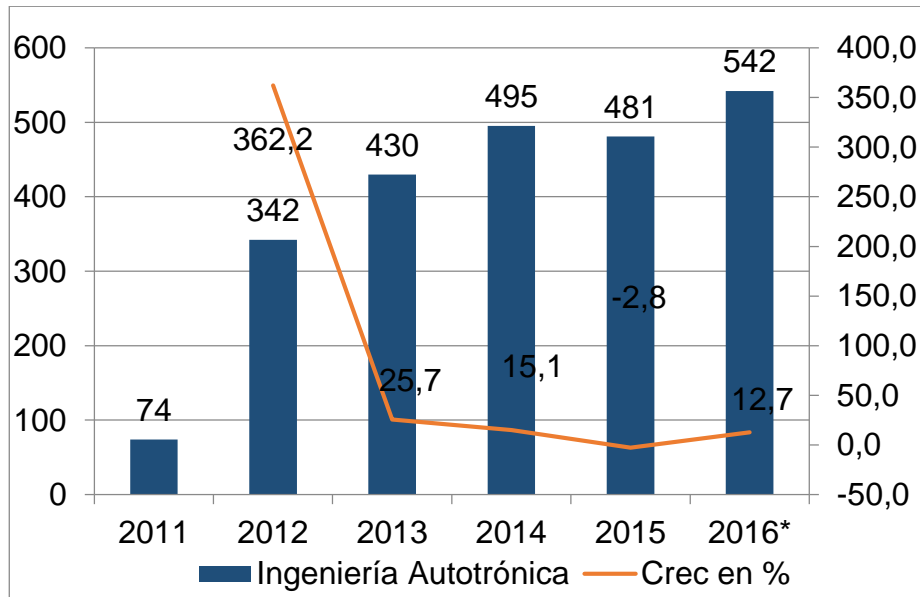
La Carrera de Ingeniería Autotrónica es de reciente creación dentro de la Universidad Pública de El Alto y del Sistema Universitario, el cual se constituye en una nueva propuesta de oferta académica para la sociedad, formando profesionales con conocimientos teórico - prácticos y capacidad de adaptación a los dinamismos tecnológicos que permitan desarrollar sus habilidades y destrezas en el campo profesional,

El egresado de esta carrera es un Técnico analítico, con espíritu emprendedor y valores personales y profesionales, altamente capacitado en el aspecto teórico-práctico, que resuelve problemas relacionados con la reparación de motores de gasolina y diésel, así como los diferentes sistemas con que cuenta el automóvil, tratando siempre de trabajar bajo el régimen de competencias cumpliendo con las expectativas que demanda la industria automotriz.

La Carrera de Ingeniería Autotrónica ha sido creada en la gestión académica 2011 con setenta y cuatro (74) estudiantes. A partir de entonces la Carrera ha tenido un

crecimiento sustancial en el número de estudiantes, tal como se observa en el gráfico N° 1:

**Figura 3.1 Detalle de estudiantes
de la carrera de ingeniería Autotrónica.**



Fuente: [Ingeniería Autotrónica,2016].

3.2.1.1 Descripción del proceso de control del personal

El proceso de control del personal administrativos, y de los docentes de ingeniería Autotrónica, empieza con la formación de todo el personal, en los diferentes horarios ya establecidos, y de esa formación se describe en detalle, todas las materias asignadas, en el registro de asistencia diaria de Docentes, en planillas llenadas de forma manual, el cual se muestra en la figura 3, el mismo que muestra los datos que deben ser llenados y firmados.

Figura 3.2 Parte diario de las actividades Ingeniería Autotronica.



REGISTRO DE ASISTENCIA DIARIA DOCENTES

FECHA: 04-NOVIEMBRE-2019 DIA: LUNES TURNO: MAÑANA

Nº	APELLIDO(S) Y NOMBRE(S)	MATERIA	SIGLA MATERIA	Nº AULA	SEMESTRE PARALELO	TEMARIO AVANZADO	HR.S. INGRESO	FIRMA	HR.S. SALIDA	FIRMA
1	PEEREZ ROMERO FELIX REYNALDO	ELECTRÓNICA DIGITAL Y LABORATORIO	ETN-301	ATN-1	3 "A"		07:30		10:30	
2	HARRIAGUE ALVAREZ MARCELO JESUS	ELECTRICIDAD AUTOMOTRIZ	ATN-403	403	4 "A"		07:30		10:30	
3	JIMENEZ CHAVEZ JUSTO EDGAR	SISTEMAS DE DIRECCIÓN, SUSPENSIÓN Y FRENOS	ATN-804	LAB-A1	6 "A"		07:30		10:30	
4	CHOQUE CORDERO WILFREDO	ELECTROMAGNETISMO Y LABORATORIO	FIS-303	ATN-2	3 "A"		10:30		13:30	
5	MAMANI CHOQUE MARCO ANTONIO	AUTOTRÓNICA II	ATN-801	ATN-1	6 "A"		10:30		13:30	

FECHA:04-NOVIEMBRE-2019 DIA: LUNES TURNO: TARDE

Nº	APELLIDO(S) Y NOMBRE(S)	MATERIA	SIGLA MATERIA	Nº AULA	SEMESTRE PARALELO	TEMARIO AVANZADO	HR.S. INGRESO	FIRMA	HR.S. SALIDA	FIRMA
1	ROJAS TINTAYA GONZALO	SISTEMAS AUTOTRÓNICOS DE VEHICULOS DE ALTO TONELAJE	ATN-801	LAB-A1	8 "A"		12:00		15:00	
2	PEEREZ ROMERO FELIX REYNALDO	MEDICIONES ELCTRÓNICAS Y LABORATORIO	ETN-100	ATN-1	1 "D"		13:30		16:30	
3	FLORES MENACHO EDWIN LUIS	AUTOTRÓNICA III	ATN-701	ATN-2	7 "A"		13:30		16:30	
4	MIRANDA MENDOZA WILSON AMADO	ELECTRÓNICA DIGITAL Y LABORATORIO	ETN-301	ATN-1	3 "B"		16:30		19:30	

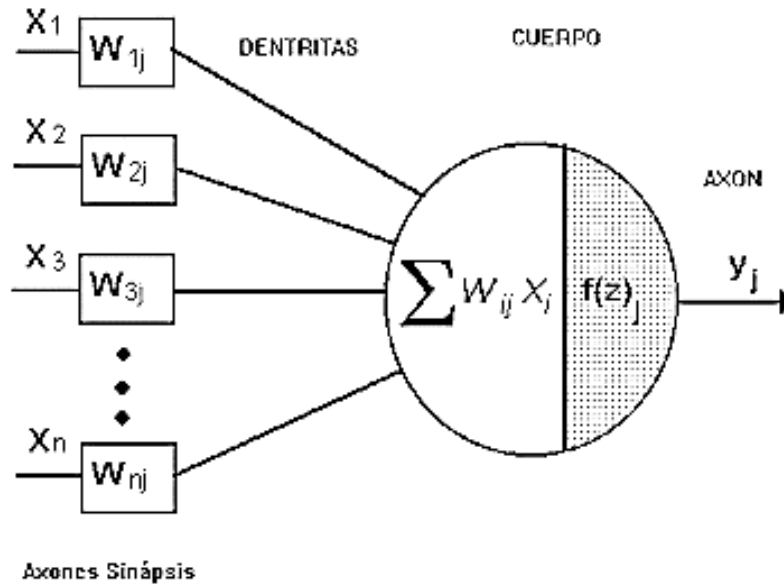
FECHA: 04-NOVIEMBRE-2019 DIA: LUNES TURNO: NOCHE

Nº	APELLIDO(S) Y NOMBRE(S)	MATERIA	SIGLA MATERIA	Nº AULA	SEMESTRE PARALELO	TEMARIO AVANZADO	HR.S. INGRESO	FIRMA	HR.S. SALIDA	FIRMA
1	VARGAS MAMANI NIXON EMILIANO	MEDICIONES ELCTRÓNICAS Y LABORATORIO	ETN-100	Nº 1	1 "B"		18:00		21:00	

Fuente: [Ingeniería Autotrónica, 2019].

3.3 Diagnóstico de la Topología y Estructura de RNA.

Figura 3.3 Estructura de la RNA.



Fuente: [Collantes, 2001].

La topología o arquitectura de una red consiste en la organización y disposición de las neuronas en la red formando capas de neuronas, en este sentido, los parámetros fundamentales para esta distinción son:

- Número de capas.
- Número de neuronas por capa.
- Grado de conectividad.
- Tipo de conexiones entre neuronas.

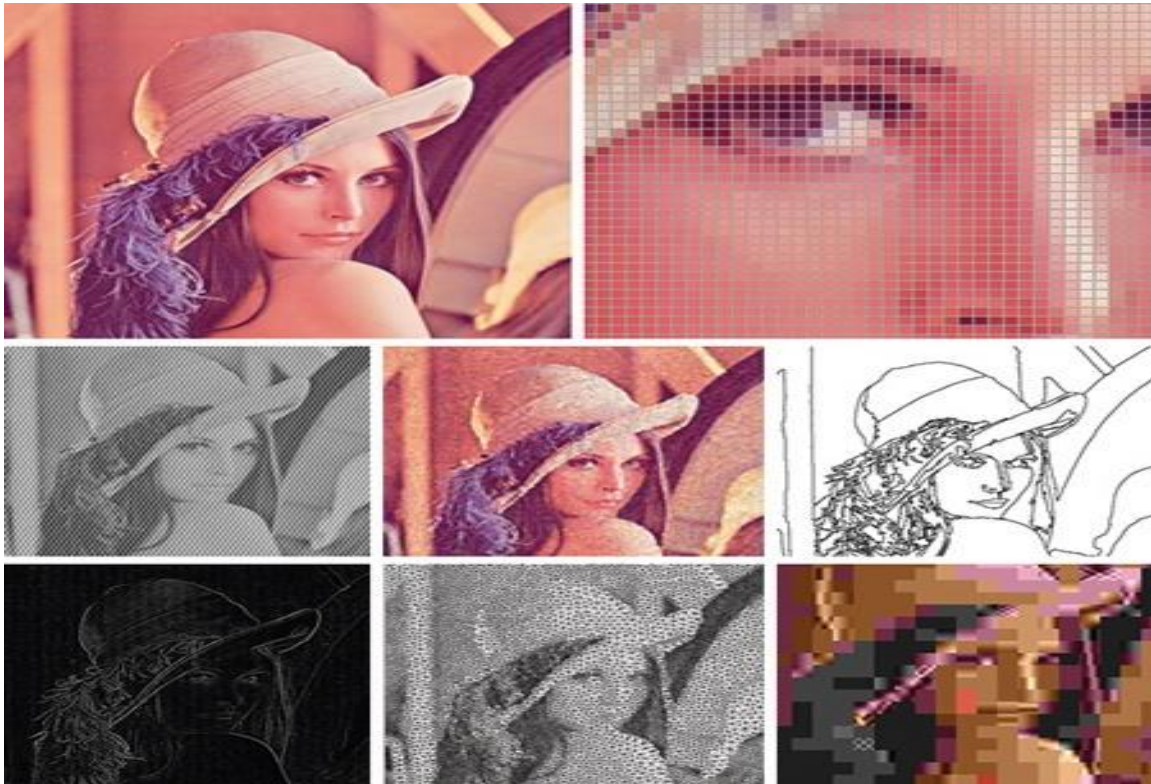
Ya que se tiene la transformada de distancia de la imagen, se sigue con el entrenamiento de una red neuronal RBF que sea capaz de aprenderla y generalizarla con un error óptimo.

Como se ha mencionado anteriormente, no existe una solución única para la arquitectura de red neuronal que se debe usar en determinado problema, el diseño de redes neuronales es más un arte que una ciencia, especialmente al escoger el número de capas ocultas, que en nuestro caso son ya definidas por las RBFs, que son de tres

capas y el número de neuronas en cada capa oculta, y las funciones de activación a usarse. Estos factores dependen enteramente del diseñador, aunque hay reglas que surgen más que nada de los resultados obtenidos en otros problemas.

3.3.1 Tratamiento de las imágenes

Figura 3.4 Tratamiento de imágenes.



Fuente: [Collantes, 2001].

El procesamiento de imágenes con MATLAB es un proceso de tres pasos en el que cargas, manipulas y muestras los resultados como salida. Si bien esto puede parecer bastante simple, muchas de las imágenes con las que trabajas requieren una manipulación precisa para obtener resultados precisos, y el proceso, así como las herramientas especializadas de procesamiento de imágenes que MATLAB proporciona, refleja este requisito.

Una vez completado el proceso, puedes realizar tareas como el análisis estadístico, la extracción de características y la medición de la propiedad con una mayor garantía de que los resultados serán correctos.

El procesamiento de imágenes complejo no es una característica incorporada en MATLAB. Mientras que puedes cargar, guardar y realizar tareas básicas de procesamiento de imágenes en el principal programa de MATLAB, las herramientas de procesamiento de imágenes adicionales hacen posible el procesamiento complejo.

Las cajas de herramientas de adquisición de imágenes o Image Processing Toolbox, a partir de la fecha de publicación, son un potente complemento a considerar.

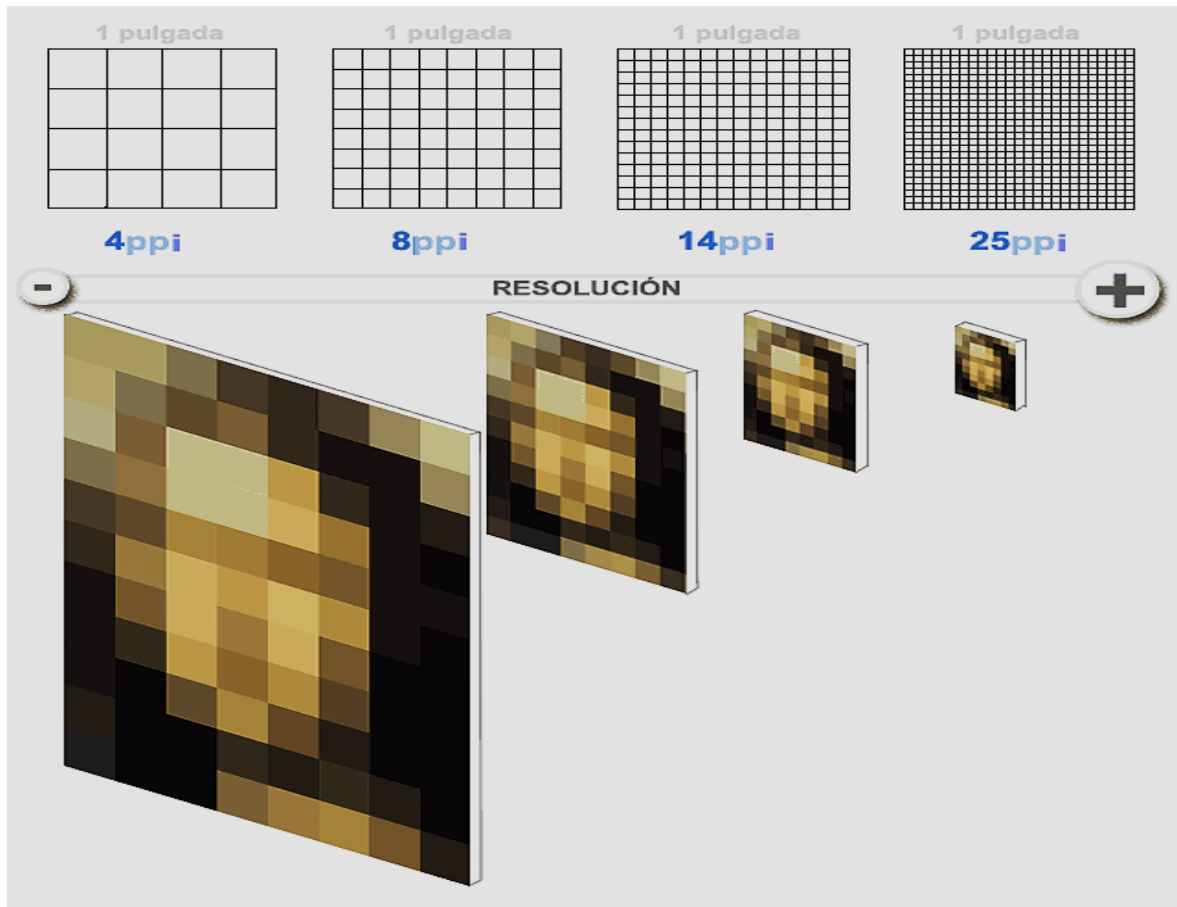
La caja de herramientas de adquisición de imágenes hace que la carga de imágenes, especialmente las que se obtienen de fuentes de alto nivel científico e industrial, sea más fácil. La caja de herramientas de procesamiento de imágenes es compatible con una amplia gama de procesos de manipulación de la imagen y la de mapeo te ayuda a convertir imágenes mapas de dos y tres dimensiones.

MATLAB y sus herramientas, en particular los de la Caja de herramientas de procesamiento de imágenes, es compatible con formatos comunes de imagen como JPEG, TIFF y PNG y los formatos de imagen menos comunes como el BIP y BIL, utilizados en imágenes de satélite y formatos especializados tales como DICOM para imágenes médicas y NITF para imágenes geoespaciales.

Las opciones de programación de MATLAB proporcionan apoyo adicional para escribir programas personalizados para manejar los formatos de imagen que MATLAB no admite directamente. Además, el tamaño de un archivo de imagen no afecta a su capacidad para trabajar y manipular debido a que MATLAB incluye los procesos de flujo de trabajo, incluyendo re muestreo espacial y el procesamiento de bloques específicos para trabajar con imágenes de gran tamaño.

3.3.1.1 Tamaño de imagen

Figura 3.5 Tamaño y resolución de una imagen.

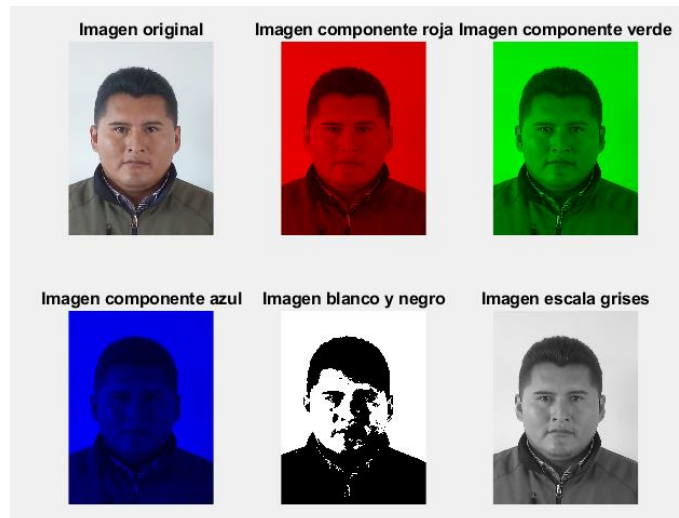


Fuente [Collantes, 2001].

La imagen de un rostro se representa utilizando un matriz, en nuestro caso la imagen tiene unas dimensiones de $(439 * 412)$ pixeles, donde $(x = 439)$ y $(y = 412)$, y al ser en RGB tendremos componentes por color, dado que conocemos el ancho y alto de la imagen el número de pixeles que obtendremos será el producto entre $((x*y) * 3)$.

3.3.1.2 Adecuación de imagen

Figura 3.6 Componentes de una imagen.



Fuente: [Elaboración propia, 2020].

Como vemos las componentes de la señal en color se obtienen simplemente obteniendo cada una de las matrices que representan cada una de las componentes del color y haciendo las demás matrices nulas, es decir, si por ejemplo queremos representar la componente roja de la imagen de color, lo que hacemos es introducir en una nueva matriz de 3 dimensiones $(m,n,3)$ (al igual que la señal de color), las filas y columnas de la primera componente de la señal de color $(m,n,1)$ en la componente 1 de nuestra nueva matriz e introducir todo ceros en las otras 2 componentes de la nueva matriz $(m,n,2$ y $m,n,3)$.

Para pasar la imagen en color a una imagen en blanco y negro, usamos el comando `im2bw`.

`IM2BW` Convertir imagen a imagen binaria de umbral.

`IM2BW` produce imágenes binarias a partir de un índice, las imágenes de intensidad, o RGB. Para hacer esto, convierte la imagen de entrada a formato de escala de grises (si no está ya una imagen de intensidad) y, posteriormente, convierte esta imagen en escala de grises a binario por umbralización. La imagen binaria de salida BW tiene

valores de 1 (blanco) para todos píxeles de la imagen de entrada con luminancia mayor que el nivel y 0 (negro) para todos los demás píxeles. (Tenga en cuenta que especifica el nivel en el intervalo [0,1], independientemente de la clase de la imagen de entrada.)

La umbralización que usa `im2bw` es una técnica para el procesado de imágenes en la que una imagen de entrada en escala de grises se re cuantifica a dos niveles de gris, o lo que es lo mismo, se convierte a imagen binaria. Cada pixel de la imagen original se compara con un umbral y el resultado de esa comparación determina si el pixel se convertirá a un nivel o a otro (blanco o negro, objeto o fondo). El algoritmo de umbralización más simple es aquél que usa un solo valor de umbral para toda la imagen.

La umbralización se suele emplear cuando hay una clara diferencia entre los objetos a extraer respecto del fondo de la escena. Al aplicar un umbral T , la imagen en escala de grises $f(x,y)$ quedará binarizada y etiquetada con '1' los píxeles correspondientes al objeto y con '0' aquellos que son del fondo. Si el umbral solo depende de $f(x,y)$ se dice que es un umbral global; en el caso de que además dependa de $p(x,y)$, por ejemplo, el valor medio de los píxeles vecinos, el umbral se denomina local; y si depende también de la posición (x,y) del pixel, se denomina dinámico.

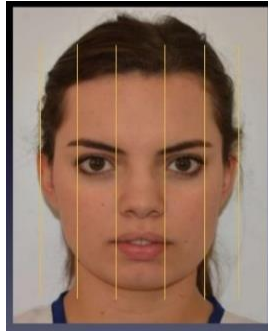
-Para pasar la imagen en color a una imagen en escala de grises, usamos el comando `rgb2gray` que lo que hace es simplemente la operación en una nueva matriz de dos dimensiones: $\text{image_Gray} = \text{image}(:, :, 1) * 0.3 + \text{image}(:, :, 2) * 0.59 + \text{image}(:, :, 3) * 0.1$.

`RGB2GRAY` Convertir imagen o mapa de colores RGB a escala de grises. `RGB2GRAY` convierte las imágenes RGB a escala de grises mediante la eliminación del tono y la saturación de información al tiempo que conserva la luminancia. $I = \text{RGB2GRAY}(\text{RGB})$ convierte la imagen de color verdadero RGB a la intensidad en escala de grises a la imagen I . $\text{NEWMAP} = \text{RGB2GRAY}(\text{MAP})$ devuelve un mapa de color en escala de grises Soporte de clase Si la entrada es una imagen RGB, puede ser `uint8`, `uint16`, `doble` o `individual`.

Ya obtenida la imagen del usuario se procede a aplicarle el método de escala de grises para poder procesar la imagen y así de esa manera poder encontrar los puntos característicos, como ser los ojos, la nariz y la boca.

3.3.2 Análisis de patrones faciales

Figura 3.7 Análisis facial (quintos).

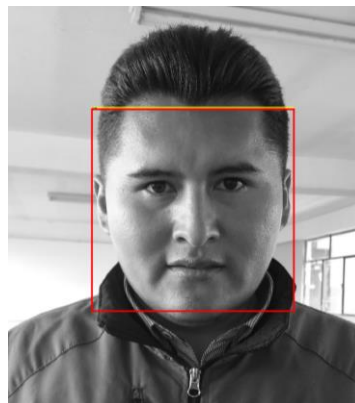


Fuente: [Serrano, 2012].

Biometría facial. Permite determinar la identidad de una persona analizando su rostro. A diferencia de otras biometrías tipo iris o huella dactilar esta tecnología no es intrusiva y no necesita de colaboración por parte del usuario. Sólo es necesario que su rostro sea adquirido por una cámara web.

3.3.2.1 Localización de los puntos característicos

Figura 3.8 Detectando rasgos faciales.



Fuente: [Elaboración propia, 2020].

Sistemas locales o geométricos: en este caso, se analizan vectores característicos extraídos del perfil del individuo que queremos estudiar, aunque también podemos comprobar los rasgos que pueden observarse de la vista frontal de la cara.

De rasgos locales: reconocen los ojos, la nariz, la boca, miden las distancias

CascadeObjectDetector detecta objetos utilizando el algoritmo de Viola-Jones

DETECTOR = vision.CascadeObjectDetector crea un objeto del sistema que detecta los objetos mediante el algoritmo de Viola-Jones.

El detector es capaz de detectar una variedad de objetos, incluyendo caras y una parte superior del cuerpo de la persona. El tipo de objeto a detectar es controlado por la propiedad ClassificationModel.

De manera predeterminada, el detector está configurado para detectar las caras y los rasgos faciales que nos interesan como ser los ojos, nariz y boca.

DETECTOR = vision.CascadeObjectDetector (MODEL) crea un objeto System, detector configurado para encontrar objetos definidos según el modelo. Modelo es una cadena que describe el tipo de objeto a detectar. Hay varias cadenas modelo válido.

Los ejemplos incluyen 'FrontalFaceCART', 'Parte superior', y 'ProfileFace'.

DETECTOR = vision.CascadeObjectDetector (XMLFILE) crea un objeto System, detector, y lo configura para utilizar el modelo de clasificación personalizado especificado con la entrada XMLFILE.

XMLFILE se puede crear utilizando la función trainCascadeObjectDetector o funcionalidad de formación OpenCV. Debe especificar una ruta completa o relativa al XMLFILE, si no está en la ruta de MATLAB.

DETECTOR = vision.CascadeObjectDetector (... , nombre, valor) configura las propiedades de los objetos del sistema, especificados como uno o más argumentos nombre-valuepair. No especificando propiedades ya que tienen valores por defecto.

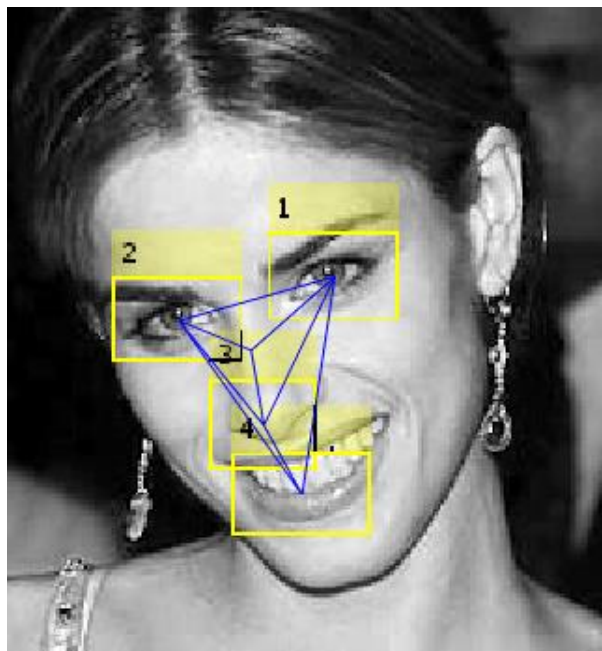
BBOXES = paso (DETECTOR, I) lleva a cabo la detección de objetos multi-escala la imagen de entrada, I, y la rentabilidad, BBOXES, una definición M-por-4 matriz cuadros delimitadores M que contienen los objetos detectados.

Cada fila de BBOXES es un vector de cuatro elementos, [x altura y anchura], que especifica la esquina superior izquierda y el tamaño de un cuadro delimitador en píxeles. Cuando no se detectan objetos, BOXES está vacía.

Debe ser una escala de grises o imagen de color verdadero (RGB).

3.3.2.2 *Posición de los ojos, nariz y boca*

Figura 3.9 Detección de características de un rostro.

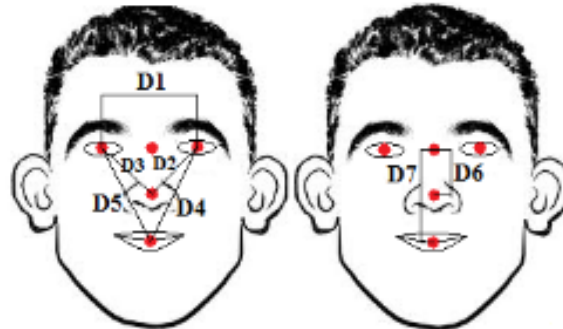


Fuente: [Elaboración propia, 2020].

Comenzando con el método de detección de ojos, La ubicación de la nariz está determinada desde la adquisición de la imagen y sus coordenadas están determinados en 'x' por la componente 'x' del punto medio y en 'y' por posición del recuadro al momento de la adquisición de la imagen.

3.3.2.3 Cálculo de distancia

Figura 3.10 Puntos encontrados en un rostro.



Fuente: [Enrique C. 2004].

Con la detección de todos los puntos del rostro, se realiza una relación entre ellos para determinar las distancias necesarias para la identificación de género como se muestra en la figura.

Se determinan relaciones de proporción entre los puntos de la imagen, con los cuales se crea la matriz con la que se entrenará el clasificador neuronal inteligente con la finalidad de realizar el reconocimiento facial biométrico. El clasificador se entrenará con la matriz de distancias, obtenidas de las imágenes tomadas de cada uno de las personas registradas en el sistema. [Enrique C.2004].

3.3.2.4 Matriz característica de usuario

La matriz característica del Usuario, es una matriz que está basada en la comparación de las distancias del usuario contra un promedio y la adición o sustracción de una desviación estándar, creando así una imagen matricial de cada usuario.

Con la matriz de Usuarios definida, se obtienen los vectores columna Promedio y Desviación estándar, los cuales son fundamentales para el desarrollo de la matriz característica, estos vectores están conformados por las filas formadas por las proporciones de las distancias obtenidas anteriormente.

Figura 3.11 Matriz de características

Obtención de los vectores Promedio y Desviación estándar.

	Usuario(1)	Usuario(2)	...	Usuario(n)				
Usuarios =	D1/	D1/	...	D1/	Promedio =	P1	Desviacion estandar =	Des1
	D2/	D2/	...	D2/		P2		Des2
	D3/	D3/	...	D3/		P3		Des3
	D4/	D4/	...	D4/		P4		Des4
	D5/	D5/	...	D5/		P5		Des5
	D6/	D6/	...	D6/		P6		Des6
	D7/	D7/	...	D7/		P7		Des7
	42*n					42		42

Fuente: [Enrique C. 2004].

Dejando claro la composición de la matriz, se llena con los valores iniciales y finales ya calculados, colocando en el intervalo cerrado [i, f].

De esta manera la imagen matricial, se representan los usuarios para el posterior análisis e identificación con la red de función base radial, de cada persona que se encuentra registrada en el sistema.

3.3.3 Descripción de las entradas

Tabla 3.1 Entradas de la red neuronal RBF.

dist	dist1	dist2	dist3	dist4	dist5	dist6	dist7	dist8	dist9
153.883	96.3872	130.078	203.476	204.822	132.524	92.2239	63.5059	145.778	82.5061
153.883	86.5708	144.941	211.568	194.589	133.686	95.1972	67.7237	139.573	72.0069
180.278	100.722	161.435	239.086	220.096	151.53	109.293	74.0017	157.351	83.7257
183.096	102.46	166.739	243.483	218.009	152.611	109.041	78	159.38	81.7435
177.703	110.293	159.958	223.788	238.354	162.929	97.7548	80.9475	159.05	79.489
158.166	95.5471	126.512	204.118	190.227	129.059	88.8046	53.0848	134.302	81.8841
138.262	81.4448	110.808	183.746	176.279	112.202	81.7221	44.1814	123.016	79.2275
158.039	94.3319	131.941	197.209	204.75	137.707	88.2511	63.5	139.032	75.5596
196.064	108.445	186.87	263.339	253.403	171.756	124.604	87.023	176.003	89.0505
169.003	98.623	144.506	212.196	220.536	151.605	92.8574	76.5408	154.137	77.6032

Fuente: [Elaboración propia, 2020].

Las distancias obtenidas del rostro de la persona, son insertadas al simulador de MatLab para redes neuronales Neural Network Toolbox, utilizando redes neuronales artificiales de función base radial.

W=

213,087423	112,94357	129,086212	262,111617	281,515985	162,026232	116,211015	60,8789783	208,572529	151,060418
------------	-----------	------------	------------	------------	------------	------------	------------	------------	------------

[Descripción de la red Base Radial

NEWRBE Diseño de una red de base radial exacta.

Son redes de base radial que se pueden usar para aproximar funciones. newrbe diseña una red de base radial con error cero en los vectores de diseño.

newrb (P, T , SPREAD) toma dos o tres argumentos:

P - matriz de vectores de entrada RXQ Q.

T - matriz SxQ de vectores de clase Q destino.

SPREAD - de las funciones de base radial, por defecto = 1.0

y devuelve una nueva red de base radial exacta.

La función `Newrb` crea iterativamente una red de base radial de una neurona a la vez. Las neuronas se añaden a la red una a una hasta que la suma del error cuadrático cae por debajo de la meta o un número máximo de neuronas ha sido alcanzado. La función `newrb` toma matrices de entrada, vectores objetivo (P, T), parámetros de diseño (meta y propagación) y devuelve la red deseada.

En cada iteración se comprueba el error de la nueva red y si baja suficiente, `newrb` finaliza; de lo contrario se agrega la siguiente neurona. Este procedimiento se repite hasta que se cumpla la meta de error o se alcanza el número máximo de las neuronas.

Es importante que el parámetro de propagación sea lo suficientemente grande para que las neuronas de base radial respondan a la superposición de las regiones del espacio de entrada, pero no tan grande para que todas las neuronas respondan de la misma manera.

Las redes de base radial tienden a tener más neuronas en la capa oculta que una red de retro propagación comparable. Esto es porque las neuronas sigmoideas pueden tener salidas sobre una gran región del espacio de entrada, mientras que las neuronas de base radial sólo responden a regiones relativamente pequeñas del espacio de entrada.

El resultado es que cuanto mayor sea el espacio de la entrada (en términos de número de entrada) más neuronas de base radial son necesarias. Por otra parte, diseñar una red de base radial a menudo toma mucho menos tiempo que la formación de una red sigmoidea lineal.

3.3.3.1 Entrenamiento de Red y resultados

Nuestra matriz de entrada será P, y cada fila de la matriz son las distancias de una cara tomada previamente, para el entrenamiento de nuestra red neuronal.

Tabla 3.2 Entrenamiento de red neuronal.

P	=									[
5,02493781	96,3872398	130,077861	203,476043	204,822484	132,523583	92,2239123	63,5059052	145,778085	82,5060604	
153,883072	86,5707803	144,941367	211,567956	194,589311	133,6862	95,1971638	67,7237034	139,572562	72,0069441	
180,277564	100,722391	161,434971	239,086281	220,096002	151,5297	109,293184	74,0016892	157,350723	83,7257428	
183,095603	102,459748	166,739317	243,483059	218,009174	152,610616	109,041277	78	159,380049	81,7435013	
177,703405	110,292792	159,957807	223,787846	238,353729	162,929433	97,7547953	80,9475139	159,050307	79,4889929	
158,166052	95,5471088	126,511857	204,118226	190,226838	129,059095	88,8045607	53,0848378	134,301899	81,8840644	
138,261528	81,4447666	110,808393	183,745748	176,278898	112,201827	81,72209	44,1814441	123,016259	79,2275205	
158,038761	94,331861	131,941275	197,208646	204,749847	137,707117	88,2510623	63,5	139,03237	75,5595791	
196,063765	108,444686	186,869607	263,33866	253,403335	171,756368	124,604374	87,0229855	176,002841	89,0505474	
169,002959	98,6230196	144,506055	212,196254	220,536278	151,604749	92,8574176	76,5408388	154,137114	77,6031572	

];

Fuente: [Elaboración Propia, 2020].

Tabla 3.3 Resultado de red neuronal.

T = [

115,632611
 129,973906
 147,751825
 149,456234
 149,026662
 126,170454
 113,088847
 129,032052
 165,655717
 139,760784

];

Fuente: [Elaboración Propia, 2020].

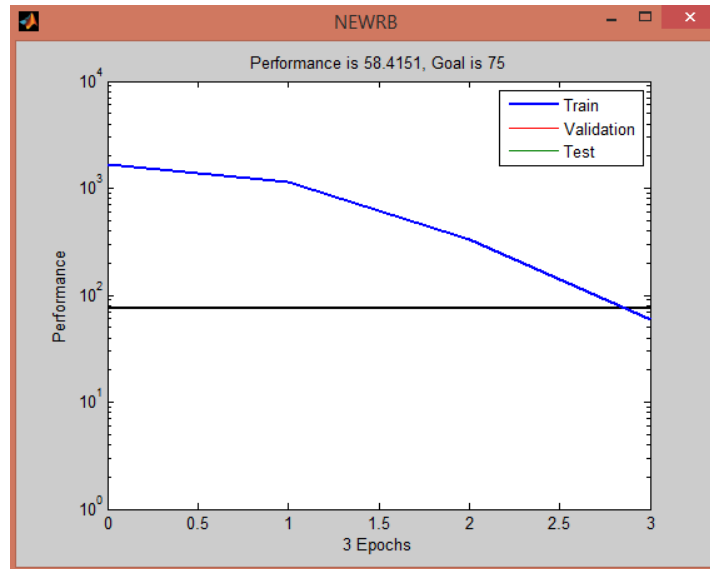
Nuestra matriz target es T, es al resultado al que se pretende llegar con el entrenamiento de nuestra red neuronal.

net = newrbn(P,T,75);

En 'net' se almacena el entrenamiento de nuestra red neuronal RBF.

Nuestro SPREAD tomará el valor = 75, y nos devolverá una nueva red de base radial exacta.

Figura 3.12 Graficando newrb (MatLab).



Fuente: [Elaboración propia, 2020].

Aquí diseñamos una red de base radial, entradas dadas X y metas T.

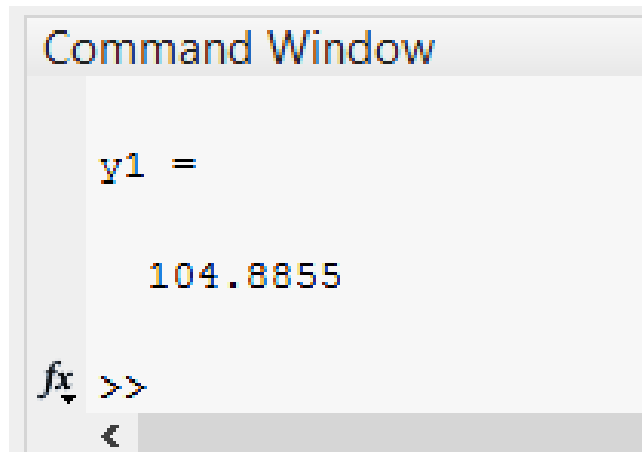
x = [

```
213,0874  
112,9436  
129,0862  
262,1116  
281,5160  
162,0262  
116,2110  
60,8790  
208,5725  
151,0604
```

];

y1= sim (net, x);

Figura 3.13 Resultado del entrenamiento a la RBF.



```
Command Window

y1 =

    104.8855

fx >>
<
```

Fuente: [Elaboración propia, 2020].

Con 'sim' simulamos nuestra red, con una nueva entrada que es 'x', y la almacenamos en 'y1'. El resultado obtenido es satisfactorio porque entra en el parámetro deseado de nuestro target 'T'.

3.4 Análisis y Diseño del Prototipo

Es claro que la Biometría ofrece tareas de identificación y verificación o autenticación de una persona, esto da una gama sin fin de aplicaciones en las cuales se necesite la presencia de un individuo en particular.

Una de esas es el control de asistencia de los empleados o trabajadores, sistema de uso general en empresas públicas y privadas.

En la mayoría de estas empresas se hace uso de sistemas electromecánicos, sistemas que no cumplen labores de verificación o autenticación; esto es un problema cuando los empleados dejan de lado sus responsabilidades y se dedican a engañar al sistema haciendo que otros compañeros o personas ajenas a la Institución registren su asistencia y por consiguiente no es sancionada por dicha inasistencia.

En todo caso la duda reinará en cuanto a la puntualidad y cumplimiento de las horas de trabajo de un empleado o trabajador mientras se usen estos sistemas electromecánicos.

Este proyecto utilizará el identificador biométrico de reconocimiento facial con la tecnología de redes neuronales de base radial (RBF). Para reemplazar la forma de control de la carrera Ingeniería Autotrónica.

3.4.1 Análisis

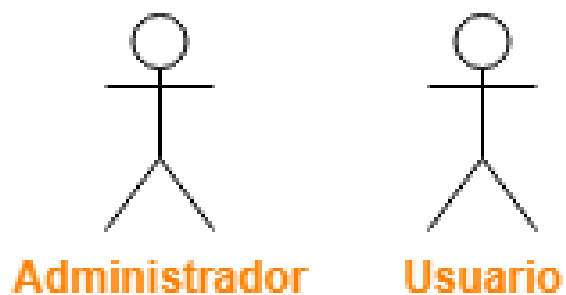
Para el análisis y diseño del Sistema biométrico facial se utilizará el estándar mundial de modelado de software UML.

3.4.2 Casos de uso

Actores del sistema

El prototipo del Sistema Biométrico Facial cuenta con 3 actores:

Figura 3.14 Casos de uso



Fuente: [Elaboración propia, 2020].

Actor Administrador:

Representa al encargado del mantenimiento de la base de datos. Su trabajo consiste en dar altas, bajas y modificaciones del personal, así como también de los cursos, requerimientos, horarios, salones, etc. Todo su trabajo podrá ser realizado a través de la aplicación, previa autenticación.

Actor Usuario:

Representa la persona que va a ingresar y debe de pasar por el sistema biométrico, para que el sistema proceda con la autenticación, están involucrados todos los Docentes de Ingeniería Autotrónica.

a) Caso de uso “AUTENTICAR SESIÓN”

Este caso de uso lo ejecuta el operador del Sistema para iniciar el funcionamiento del Sistema Biométrico Facial, el operador tiene que estar autorizado.

a.1 Flujo de eventos

Flujo Básico.

Tabla 3.4 Flujo de eventos a.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador inicia la ventana de la aplicación, apertura una ventana que le solicitará ingreso de clave personal.	
2. El administrador ingresa los datos para iniciar la sesión.	
3. El administrador envía los datos para que el sistema lo valide.	
	4. El sistema procesa la información, realizando una búsqueda de administrador autorizado.
	5. El sistema muestra un mensaje con la pantalla de la sesión de trabajo (pantalla principal).

Fuente: [Elaboración propia, 2020].

a.1. Flujo Alternativo.

- El administrador puede elegir salir del sistema, debido a problemas con la clave personal o el sistema le solicitará un usuario y contraseña válidos.

- El administrador puede decidir no enviar los datos para la validación y salir del sistema.
- Problemas con la conexión con la Base de datos.
administrador no existe.
- Los datos del administrador no son válidos.

a.2. Requisitos Especiales

El administrador debe al menos tener conocimiento de informática, de entorno de ventanas gráficas.

a.3. Pre Condiciones

El administrador tiene que tener su clave personal designada para su entrada en el sistema.

a.4. Post Condiciones

El sistema dará como resultado un mensaje de éxito, que le solicitará su confirmación para el acceso al sistema

a.5. Puntos de extensión.

Este caso de uso no cuenta con casos de uso extendidos, por tanto, no tiene puntos de extensión.

b) Caso de uso “REGISTRAR SESIÓN”

Este caso de uso lo ejecuta el administrador que una vez autenticado se procederá a registrar sus datos y se iniciará el funcionamiento del Sistema para el Control de Docentes, de Ingeniería Autotrónica, perteneciente al Área Desarrollo tecnológico productivo, UPEA.

b.1. Flujo de eventos
Flujo Básico

Tabla 3.5 Flujo de eventos b.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador confirma que va iniciar la sesión para el control de personal.	
	2. El sistema procede a la validación de los datos.
	3. Se confirma la autenticidad de los datos del operador.
	4. El sistema autoriza y se muestra la ventana de control de personal.

Fuente: [Elaboración propia, 2020].

b.1. Flujo Alternativo

El sistema comunica al administrador que hay problemas de que no está autorizado por el Sistema para iniciar la sesión de control de personal.

b.2. Requisitos Especiales

El administrador debe al menos tener conocimiento de informática, de entorno de ventanas gráficas.

b.3. Pre Condiciones

El administrador debe haber puesto su clave de autorización para la búsqueda de la misma en el sistema.

b.4. Post Condiciones

El sistema dará como resultado un mensaje de éxito, y mostrará la ventana de control de personal.

b.5. Puntos de extensión.

Este caso de uso no cuenta con casos de uso extendidos, por tanto, no tiene puntos de extensión.

c) Caso de uso “AUTENTICAR USUARIO”

Este caso de uso lo ejecuta el usuario del Sistema, acercando su rostro a la cámara web, luego el sistema validará su rostro para compararla con la base de datos.

c.1. Flujo de eventos.

Flujo Básico.

Tabla 3.6 Flujo de eventos c.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
	1. El Sistema está a la espera del rostro de un usuario.
2. Un usuario pone su rostro cerca de la cámara web.	
	3. El Sistema captura ese rostro y la procesa y la envía para la comparación con base de datos.
	4. El rostro buscado es válido, entonces comunicará que el usuario es válido.

Fuente: [Elaboración propia, 2020].

c.1. Flujo Alternativo.

- El usuario puede haber tenido un accidente (cortes, heridas, hinchazones etc.)

- El sistema no establece conexión con la base de datos, por errores del sistema.
- El rostro no fue encontrado en la base de datos, entonces se retorna el mensaje de que el usuario no está registrado.

c.2. Requisitos Especiales.

El administrador debe al menos tener conocimiento de informática, de entorno de ventanas gráficas.

c.3. Pre Condiciones.

- Para realizar este caso de uso, el administrador del sistema debe haber iniciado con éxito la sesión de control de personal.
- El usuario (Docente) tiene que estar dado de alta en el Sistema.
- Los usuarios deben ser únicamente de los Docentes de Ingeniería Autotrónica.

c.4. Post Condiciones.

El sistema dará como resultado los datos del usuario (fotografía y otros datos personales).

c.5. Puntos de extensión.

Este caso de uso no cuenta con casos de uso extendidos, por tanto, no tiene puntos de extensión.

d) Caso de uso “REGISTRAR ACCESO”

Este caso de uso lo ejecuta el usuario del sistema, una vez validada el rostro del usuario, el sistema procederá a registrar los datos obtenidos.

d.1. Flujo de eventos.
Flujo Básico.

Tabla 3.7 Flujo de eventos d.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
	1. El Sistema retorna el mensaje del rostro válido. Con los datos del usuario.
	2. El sistema registra los datos en la base de datos de forma automática.
3. El usuario ingresa los parámetros correctamente, y son guardados en la base de datos.	
	4. El Sistema le solicita reiniciar

Fuente: [Elaboración propia, 2020].

d.1.Flujo Alternativo.

- El Sistema retorna el mensaje de usuario no encontrado, para que el usuario se registre.
- El Usuario decide cancelar su acceso al sistema.
- El Sistema retorna el mensaje de que los datos enviados no son correctos, que por favor envíe datos nuevamente.

d.2. Requisitos Especiales.

El usuario debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

d.3. Pre Condiciones.

- Para realizar este caso de uso, debe haber pasado el proceso de autenticar el acceso del rostro ingresado.

- Usuario tiene que estar de alta en el sistema.

d.4. Post Condiciones.

El sistema dará como resultado un mensaje de éxito, y se registrará un nuevo acceso en el Sistema.

d.5. Puntos de extensión.

Este caso de uso no cuenta con casos de uso extendidos, por tanto, no tiene puntos de extensión.

e) Caso de uso “LISTADO DE ASISTENCIA”

Este caso de uso lo ejecuta el supervisor del sistema y le permite consultar las asistencias de los de los docentes.

e.1. Flujo de eventos.

Flujo Básico.

Tabla 3.8 Flujo de eventos e.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador solicita listado de asistencia de personal.	
2. El administrador, selecciona ciertos parámetros para la solicitud.	
3. El administrador envía la solicitud.	
	4. El Sistema atiende la solicitud y realiza una búsqueda en la base de datos de la información solicitada por el administrador.

	5. Se envía los resultados de la búsqueda al administrador en forma de reporte.
--	---

Fuente: [Elaboración propia, 2020].

e.1. Flujo Alternativo.

- El administrador envía la solicitud con los parámetros por defecto para su búsqueda en el sistema.
- El Sistema retorna el mensaje de que no se pudo establecer conexión con la base de datos.

e.2. Requisitos Especiales.

El usuario debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

e.3. Pre Condiciones

- El administrador debe haberse validado al entrar al sistema con su login y password (medida de contingencia) y además debe estar de alta en el sistema.
- El administrador debe haber seleccionado de su pantalla actual la opción de Listar Asistencia.

e.4. Post Condiciones

El sistema dará como resultado una pantalla de informe con opción de imprimir el reporte.

f) Caso de uso “RESUMEN DE ASISTENCIA DEL PLANTEL DOCENTE”

Este caso de uso lo ejecuta el administrador del sistema y le permite tener un listado de las faltas y retrasos que registra el sistema biométrico facial de los Docentes de Ingeniería Autotrónica.

f.1. Flujo de eventos.

Flujo Básico.

Tabla 3.9 Flujo de eventos f.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador solicita el resumen de asistencia de personal.	
2. El administrador, selecciona ciertos parámetros para obtener un resumen personalizado (faltas o retrasos).	
3. El administrador envía la solicitud.	
	4. El Sistema atiende la solicitud y realiza una búsqueda en la base de datos de la información solicitada del usuario.
	5. Se envía los resultados de la búsqueda al administrador en forma de reporte.

Fuente: [Elaboración propia, 2020].

f.1. Flujo Alternativo.

- El administrador envía la solicitud con los parámetros por defecto para su búsqueda en el sistema.
- El Sistema retorna el mensaje de que no se pudo establecer conexión con la base de datos.

f.2. Requisitos Especiales.

El administrador debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

f.3. Pre Condiciones.

- El administrador debe haberse validado al entrar al sistema con su login y password (medida de contingencia) y además debe estar de alta en el sistema.
- El administrador debe haber seleccionado de su pantalla actual la opción de Resumen de Asistencia de Personal.

f.4. Post Condiciones.

El sistema dará como resultado una pantalla de informe con la opción de imprimir el reporte.

f.5. Puntos de extensión.

Este caso de uso cuenta con el caso de uso “imprimir”, el cual da la opción enviar la información para imprimir.

g) Caso de uso “IMPRIMIR”

Este caso de uso lo ejecuta el administrador del sistema y es utilizado para la impresión de reportes del sistema.

g.1. Flujo de eventos.

Flujo Básico.

Tabla 3.10 Flujo de eventos g.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador ENVÍA la impresión del reporte.	
	2. El Sistema atiende la solicitud y envía la solicitud a la impresora.
	3. El sistema comunica que la impresión fue terminada con éxito.

Fuente: [Elaboración propia, 2020].

g.1. Flujo Alternativo.

- El sistema comunica al administrador que la impresora está desconectada o le falta papel.
- El Sistema retorna el mensaje de que no se pudo establecer conexión con la impresora o hubo problemas.

g.2. Requisitos Especiales.

El administrador debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

g.3. Pre Condiciones

- El administrador debe haberse validado al entrar al sistema con su login y password (medida de contingencia) y además debe estar de alta en el sistema.
- El administrador debe haber seleccionado de su menú principal cualquier opción de consulta de información para visualizar esta información.

g.4. Post Condiciones

El sistema dará como resultado una pantalla que le informará que se imprimió con éxito el reporte.

g.5. Puntos de extensión.

Este caso de uso cuenta con el caso de uso "imprimir", el cual da la opción enviar la información para imprimir.

h) Caso de uso "ENTRENAR RBF"

Este caso de uso lo ejecuta el administrador del Sistema para iniciar el entrenamiento de la red neuronal artificial del Sistema Biométrico Facial, el administrador tiene toda la autorización.

h.1 Flujo de eventos

Flujo Básico

Tabla 3.11 Flujo de eventos h.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador inicia la ventana del sistema, apertura una ventana que le solicitará ingreso de clave personal.	
2. El administrador ingresa los datos para iniciar la sesión.	
3. El administrador ingresa los datos para que para que la RBF sea entrenada y el sistema lo valide.	
	4. El sistema procesa la información, entrenando la RBF
	5. El sistema muestra los datos obtenidos del entrenamiento de la RBF (pantalla principal).
6. el administrador ingresa los datos obtenidos en el sistema principal, para el reconocimiento facial.	

Fuente: [Elaboración propia, 2020].

h.1. Flujo Alternativo

- El administrador puede elegir salir del sistema, debido a problemas con la clave personal o el sistema le solicitará un usuario y contraseña válidos.
- El administrador puede decidir no enviar los datos obtenidos y salir del sistema.

- Problemas con los datos obtenidos del entrenamiento.
- Los datos del administrador no son válidos (no existe).

h.2. Requisitos Especiales

El administrador debe tener conocimiento avanzados de informática e inteligencia artificial.

h.3. Pre Condiciones

El administrador tiene que tener su clave personal para la entrada en el sistema.

h.4. Post Condiciones

El sistema dará como resultado un mensaje de éxito, que le solicitará su confirmación para el acceso al sistema

h.5. Puntos de extensión.

Este caso de uso no cuenta con casos de uso extendidos, por tanto, no tiene puntos de extensión.

i) Caso de uso “DAR ALTA PERSONAL”

Este caso de uso lo ejecuta el administrador del sistema y le permite autorizar o habilitar el acceso al sistema.

i.1. Flujo de eventos.

Flujo Básico.

Tabla 3.12 Flujo de eventos i.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El Administrador solicita la opción de dar de Alta o Autorizar al Personal.	
	2. El Sistema le muestra una pantalla que le solicita los datos del personal a modificar.

3. El Administrador digita los datos del Personal y acepta.	
	4. El Sistema muestra la información de Personal seleccionado.
5. El Administrador procede a modificar el estado del Personal a Dar de Alta y confirma los cambios.	
	6. Muestra una pantalla de éxito en la operación

Fuente: [Elaboración propia, 2020].

i.1. Flujo Alternativo.

- El Administrador busca al personal y se da cuenta que no lo tiene registrado.
- El Administrador decide cancelar la opción de Dar de Alta y sale de esta opción.
- El Sistema muestra un aviso al administrador donde le comunica que hubo problemas con el registro de dar Alta al Personal.

i.2. Requisitos Especiales.

El usuario debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

i.3. Pre Condiciones.

- El Administrador debe haberse validado con su contraseña.
- El administrador debe haber seleccionado la opción de gestión de personal.
- El Administrador debe haber seleccionado la opción de dar de alta personal.
- El Administrador solicita buscar datos del Personal en el sistema para poder darlo de alta.

i.4. Post Condiciones.

En el sistema se generará una orden de alta de Personal, entonces se dará de alta al personal, con disponibilidad de acceso al Sistema.

i.5. Puntos de extensión.

No existen puntos de extensión.

j) Caso de uso “DAR BAJA PERSONAL”

Este caso de uso lo ejecuta el administrador del sistema y le permite desautorizar o dar de baja o deshabilitar el acceso al sistema.

j.1. Flujo de eventos.

Flujo Básico.

Tabla 3.13 Flujo de eventos j.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador solicita la opción de dar de Baja al Personal.	
	2. El Sistema le muestra una pantalla que le solicita los datos del personal a modificar.
3. El administrador digita los datos del Personal y acepta.	
	4. El Sistema muestra la información de Personal seleccionado.
5. El administrador procede a modificar el estado del Personal a Dar de Baja y confirma los cambios.	

	6. Muestra una pantalla de éxito en la operación.
--	---

Fuente: [Elaboración propia, 2020].

Flujo Alternativo.

- El administrador busca al personal y se da cuenta que no lo tiene registrado.
- El administrador decide cancelar la opción de Dar de Baja y sale de esta opción.

j.2. Requisitos Especiales.

El usuario debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

j.3. Pre Condiciones.

- El Administrador debe haberse validado con su contraseña.
- El administrador debe haber seleccionado la opción de gestión de personal.
- El Administrador debe haber seleccionado la opción de dar de baja personal.
- El Administrador solicita buscar datos del Personal en el sistema para poder darlo de baja.

j.4. Post Condiciones.

En el sistema se generará una orden de baja de Personal, entonces se dará de baja al personal, sin disponibilidad de acceso al Sistema.

j.5. Puntos de extensión.

No existen puntos de extensión.

k) Caso de uso “MODIFICAR PERSONAL”

Este caso de uso lo ejecuta el administrador del sistema y le permite modificar datos de los Docentes de Ingeniería Autotrónica.

k.1. Flujo de eventos.

Flujo Básico.

Tabla 3.14 Flujo de eventos k.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador solicita la opción de Modificar.	
	2. El Sistema le muestra una pantalla que le solicita los datos del personal a
3. El administrador digita los datos del Personal y acepta.	
	4. El Sistema muestra la información de Personal seleccionado.
5. El administrador procede a modificar los datos del personal y confirma los cambios.	
	6. Muestra una pantalla de éxito en la operación.

Fuente: [Elaboración propia, 2020].

Flujo Alternativo.

- El Administrador busca al personal y se da cuenta que no lo tiene registrado.
- El Administrador decide cancelar la opción de Modificar datos y sale de esta opción.
- El Administrador muestra un aviso que hubo problemas con el registro de la

Modificación del Personal.

k.2. Requisitos Especiales.

El usuario debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

k.3. Pre Condiciones.

- El Administrador debe haberse validado con su contraseña.
- El administrador debe haber seleccionado la opción de gestión de personal.
- El Administrador debe haber seleccionado la opción de modificar datos del personal.
- El Administrador solicita buscar datos al Personal en el sistema para poder modificar sus datos.

k.4. Post Condiciones.

En el sistema se generará una modificación del Personal, un nuevo cambio se guardará en el sistema.

k.5. Puntos de extensión.

No existen puntos de extensión.

I) Caso de uso “BUSCAR PERSONAL”

Este caso de uso lo ejecuta el administrador del sistema y le permite realizar la búsqueda de los Docentes de Ingeniería Autotrónica.

I.1. Flujo de eventos.

Flujo Básico.

Tabla 3.15 Flujo de eventos I.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador digita los datos del personal del docente que se busca	

	2. El sistema muestra los datos de este docente
3. El administrador selecciona y acepta al personal.	

Fuente: [Elaboración propia, 2020].

Flujo Alternativo.

- El Administrador se da cuenta que el Personal buscado no está en la Base de Datos.

I.2. Requisitos Especiales.

El usuario debe al menos tener conocimiento de informática (Windows), entorno de ventanas gráficas.

I.3. Pre Condiciones.

- El Administrador debe haberse validado con su contraseña.
- El administrador debe haber seleccionado la opción de gestión de personal.
- El administrador debe haber seleccionado la opción de buscar personal.

I.4. Post Condiciones.

El sistema visualizara la pantalla con los datos del personal seleccionado.

I.5. Puntos de extensión.

No tiene puntos de extensión.

m) Caso de uso “REGISTRAR ASISTENCIA DE PERSONAL”

Este caso de uso lo ejecuta el Administrador del Sistema y le permitirá ingresar la asistencia manual del personal, esto se debe a problemas que se presenten en el Sistema (es utilizado como medida de contingencia).

m.1. Flujo de eventos.

Flujo Básico.

Tabla 3.16 Flujo de eventos m.

ACCIÓN DE ACTORES	RESPUESTA DEL SISTEMA
1. El administrador selecciona la opción de Registrar Asistencia de Personal	
	2. El Sistema muestra la pantalla de Registro de Asistencia de Personal.
3. El administrador selecciona los datos del personal para registrar su acceso al sistema.	
4. El administrador selecciona si va ser registro de entrada o de salida.	
5. El administrador llena los datos que corresponden según sea entrada o salida de los Docentes de Ingeniería Autotrónica.	
6. El administrador decide guardar los datos.	

	7. El Sistema le muestra un mensaje que los datos ingresados fueron guardados con éxito.
--	--

Fuente: [Elaboración propia, 2020].

Flujo Alternativo.

- El Administrador no encuentra los datos del personal en el Sistema.
- El Administrador puede decidir no enviar los datos y salir del sistema.
- Problemas con la conexión con la Base de datos y el Sistema muestra un mensaje de que hubo problemas con el registro de los datos del personal.

m.2. Requisitos Especiales.

El Administrador debe al menos tener conocimiento de informática, de entorno de ventanas gráficas.

m.3. Pre Condiciones.

- El Administrador ha pasado la prueba de validación para el acceso al Sistema.
- El Administrador ha seleccionado previamente en el menú la opción de Registrar Asistencia de Personal.

m.4. Post Condiciones.

El sistema dará como resultado un mensaje de éxito, y mostrará la ventana de control de personal.

m.5. Puntos de extensión.

Este caso de uso no cuenta con casos de uso extendidos, por tanto, no tiene puntos de extensión.

3.4.3 Diagrama de secuencias

Los diagramas de secuencia se encuentran desarrollados en anexos.

3.4.4 Diseño

En el diseño del sistema biométrico de reconocimiento facial intervendrán tantos componentes de Hardware y de Software, en este sentido y para una mejor explicación el diseño se dividió en estos dos elementos. Los cuales fueron desarrollados con anterioridad en el capítulo 1.

3.4.5 Diagrama de interfaz

Es la representación visual disponible en donde interactúa usuario con el software, es decir, un entorno gráfico dispuesto para trabajo del usuario en donde se obtiene reconocimiento de rostro y registro de horarios.

En la Figura se muestra la propuesta de la interface para reconocimiento de rostro del usuario la misma que fue diseñada con la herramienta de MatLab GUIDE. (Nota: la imagen del usuario será incluida si previamente el usuario se encuentra registrado en el sistema).

Autenticar Usuario

Se solicitará las credenciales del usuario para poder ingresar nuevos usuarios al sistema de escritorio, estos serán comparados con el registro de la base de datos del sistema.

Registrar Usuario

Será incluido la imagen del usuario si previamente se encuentra registrado en sistema de reconocimiento facial con RBF

Reconocer Usuario

El usuario se posicionará frente a la cámara y será reconocido por el sistema, que trabajará con redes neuronales artificiales RBF, tomando como datos las distancias de los ojos, nariz y boca. Si previamente está registrado, en el sistema.

3.5 Validación del Sistema de Reconocimiento Facial

3.5.1 Matriz de confusión

En el campo de la inteligencia artificial una matriz de confusión es una herramienta que permite la visualización del desempeño de un algoritmo que se emplea en aprendizaje supervisado. Cada columna de la matriz representa el número de predicciones de cada clase, mientras que cada fila representa a las instancias en la clase real. Uno de los beneficios de las matrices de confusión es que facilitan ver si el sistema está confundiendo dos clases.

3.6 Demostración del Prototipo

3.6.1 Implementación

Una vez realizados las fases de especificación de requisitos, diseño del software, la codificación y las pruebas, se implementa el prototipo:

Captura de Ventanas del prototipo:

- a) Pantalla de Inicio de sesión para acceso al prototipo, usuario: admin , contraseña:Admin

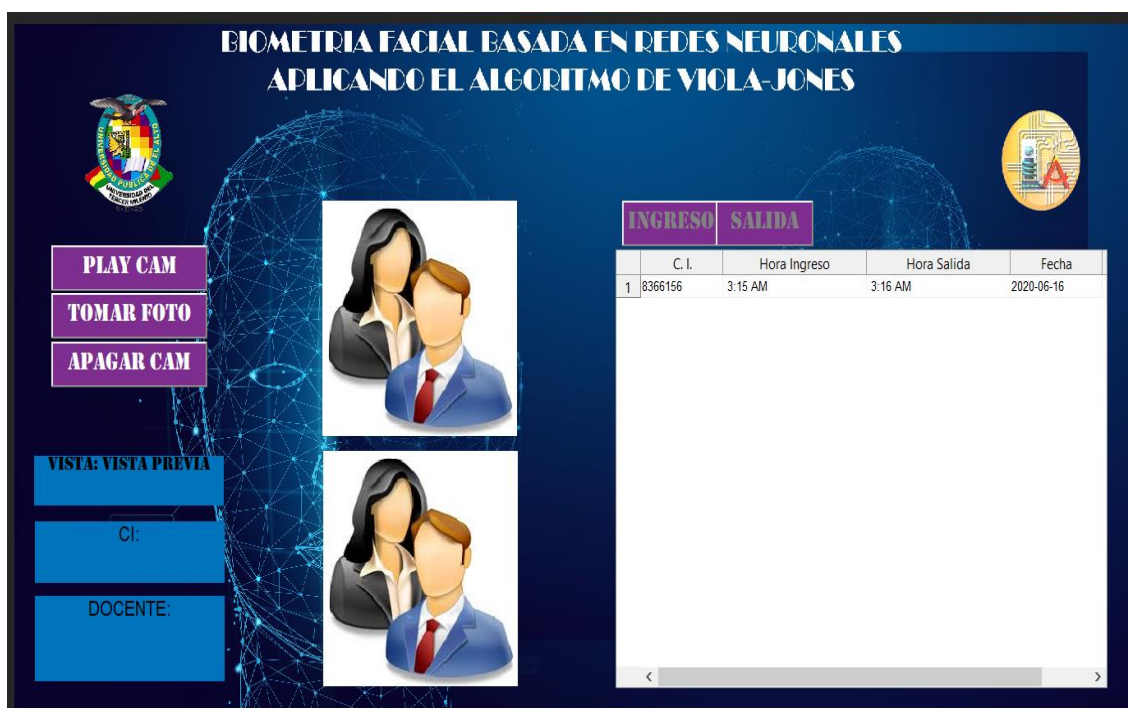
Figura 3.15 Inicio de sesion.



Fuente: [Elaboración Propia, 2020].

b) Pantalla principal del sistema.

Figura 3.16 Pantalla principal.



Fuente: [Elaboración Propia, 2020].

c) Listar Usuarios

Figura 3.17 Pantalla principal.

LISTA DE USUARIOS

ID USU...	USUARIO	PASSWORD
1	1 admin	Admin

Fuente: [Elaboración Propia, 2020].

d) Nuevo docente del sistema

Figura 3.18 Registro de docente nuevo.

REGISTRO DE DOCENTE NUEVO

16-Jun-2020 11:28 AM

PLAY CAM

TOMAR FOTO

APAGAR CAM

VISTA PREVIA






Cédula de Identidad:

Nombres:

Apellido Paterno:

Apellido Materno:

Sexo:

Fecha Nacimiento:

Estado Civil:

Profesión:

Dirección:

GUARDAR **CANCELAR**

ID	CI	APELLIDO PATERNO	APELLIDO MATERNO	NOMBRES	PROFESION	FECHA REGISTRO
1	18366156	MACHACA	TICONA	BLANCA STEFANI	ING. DE SISTEMAS	2020-06-16

Fuente:[Elaboracion Propia, 2020].

e) Perfil de docente del sistema

Figura 3.19 Perfil de Docente.

PERFIL DE DOCENTE

Cédula de Identidad: 8366156 **BUSCAR**

Cédula de Identidad:	8366156
Nombres:	BLANCA STEFANI
Apellido Paterno:	MACHACA
Apellido Materno:	TICONA
Sexo:	FEMENINO
Fecha Nacimiento:	1992-11-03
Estado Civil:	SOLTERO/A
Profesión:	ING. DE SISTEMAS
Dirección:	LORETO

CANCELAR

Imágenes: Foto de la docente y una imagen de interfaz de usuario con términos como DATA, INFORMACIÓN, BACKUP, AI, CLOUD, y WEB.

Fuente:[Elaboracion Propia, 2020].

f) Entrenamiento de la red neuronal

Figura 3.20 Entrenamiento.

ENTRENAMIENTO

Cédula de Identidad: 1
8366156 **BUSCAR**

No. Entrenamiento: [Barra verde]

PLAY CAM
TOMAR FOTO
APAGAR CAM

Docente: MACHACA TICONA BLANCA STEFANI

GUARDAR **CANCELAR**

Imágenes: Foto de la docente, iconos de una pareja, y un fondo con términos como AI, DATA, INFORMACIÓN, BACKUP, CLOUD, y WEB.

Fuente:[Elaboracion Propia, 2020].

g) Reporte diario

Figura 3.21 Reporte Diario.

	A	B	C	D	E	F	G	H
1			REPORTE	DIARIO				
2		Id	C.I.	Entrada	Salida	Fecha		
3		1	8366156	3:15 AM	3:16 AM	16/06/2020		
4								

Fuente:[Elaboracion Propia, 2020].

h) Reporte por fechas

Figura 3.22 Reportes por Fechas.

REPORTES

Fecha Inicio: >> 13-Jun-2020

A

Fecha Fin: >> 23-Jun-2020

Buscar

Excel

CANCELAR

Id Doc...	Hora Ingreso	Hora Salida	Fecha
1	3:15 AM	3:16 AM	2020-06-16

REGISTRO DE ASISTENCIA DIARIA DOCENTES
CARRERA: INGENIERIA AUTOTRONICA

	DE:	13/06/2020		
FECHA: martes, 16 de junio de 2020	A:	23/06/2020		
N	ID	HORA INGRESO	HORA SALIDA	FECHA
1	1	3:15 AM	3:16 AM	16/06/2020
2				
3				
4				

Fuente:[Elaboracion Propia, 2020].

4. MÉTRICAS DE CALIDAD Y ESTIMACIÓN DE COSTOS

4.1 Métrica de Calidad del Software ISO/IEC 9126

La métrica de calidad ISO/IEC 9126, clasifica la calidad del software en un conjunto estructurado de características y sub características de la siguiente manera:

4.1.1 Funcionalidad

Un conjunto de atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen las necesidades implícitas o explícitas.

Tabla 4.1 Ponderación de funcionabilidad.

Característica	Ponderación
Adecuación	80%
Exactitud	90%
Interoperabilidad	95%
Seguridad	90%
Cumplimiento funcional	95%
Total:	90%

Fuente: [Elaboración propia en base a ISO 9126].

4.1.2 Fiabilidad

Un conjunto de atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período establecido. Para el cálculo de la fiabilidad, se procede a plantear la siguiente fórmula (1), en base a cálculos efectuados en Cocomo II:

$$\text{Fiabilidad} = 1 - (\text{numero de errores} / \text{numero de lineas de codigo}) \quad (1)$$

$$\text{Fiabilidad} = 1 - (5 / 19625)$$

$$\text{Fiabilidad} = 0.99 * 100\% = 99\%$$

4.1.3 Usabilidad

Un conjunto de atributos relacionados con el esfuerzo necesario para su uso, y en la valoración individual de tal uso. Se realiza una evaluación del Modelo Predictivo en base a las siguientes características:

Tabla 4.2 Ponderación de usabilidad.

Característica	Ponderación
Aprendizaje: interfaz de usuario amigable	90%
Comprensión	93%
Operatividad	91%
Atractividad	90%
Total:	91%

Fuente: [Elaboración propia en base a ISO 9126].

4.1.4 Mantenibilidad

Conjunto de atributos relacionados con la facilidad de extender, modificar o corregir errores en un sistema software. Se plantea la siguiente fórmula:

$$\text{Mantenibilidad} = (M_t - (F_c + F_a + F_d)) / M_t$$

Donde:

M_t = número de módulos en la versión actual.

F_c = número de módulos en la versión actual que han cambiado.

F_a = número de módulos en la versión actual añadido.

F_d = número de módulos en la versión anterior que se ha borrado.

Entonces:

$$M_t = 3; F_c = 0; F_a = 0; F_d = 0$$

$$\text{Mantenibilidad} = (3 - (0 + 0 + 0)) / 3$$

$$\text{Mantenibilidad} = 1 * 100\% = 100\%$$

4.1.5 Eficiencia

Conjunto de atributos relacionados con la relación entre el nivel de desempeño del Modelo Predictivo y la cantidad de recursos necesarios bajo condiciones establecidas. Se utiliza la siguiente fórmula:

$$\text{Eficiencia del Software} = \text{Eficiencia}/\text{LCC}$$

Donde:

$$\text{Eficiencia} = \text{disponibilidad} * \text{confiabilidad} * \text{mantenibilidad} * \text{capacidad}$$

LCC = Costo de ciclo de vida

La disponibilidad es una medida frecuente en que el prototipo está listo para operar, para ello una estimación propia de la disponibilidad es de un 90%.

La capacidad es de un 90%, valorando que el modelo predictivo es capaz de desempeñar su función.

El costo del ciclo de vida LCC se considera un 88%.

Reemplazando en la fórmula de la eficiencia del software, se tiene:

$$\text{Eficiencia del Software} = (0.9 * 0.99 * 1 * 0.9)/0.88$$

$$\text{Eficiencia del Software} = 0.91 * 100\% = 91\%$$

4.1.6 Portabilidad

Conjunto de atributos relacionados con la capacidad para ser transferido y adaptado desde una plataforma a otra.

$$\text{Portabilidad} = 1 - (\text{ndpm}/\text{ndim})$$

ndpm = número de días para portar el modelo, valor estimado = 0.5 días.

ndim = número de días para implementar el modelo, valor estimado = 3 días.

$$\text{Portabilidad} = 1 - (0.5/3)$$

$$\text{Portabilidad} = 0.83 * 100\% = 83\%$$

4.1.7 Análisis de resultados

Se ha calculado de manera independiente cada una de las características de alto nivel de la norma ISO 9126, los resultados encontrados sirven para realizar el cálculo de la calidad global del Modelo Predictivo, tal como se muestra en la *Tabla N° 4.3*.

Tabla 4.3 Resultados de características de alto nivel ISO 9126.

N°	Característica	Preferencia	% de Preferencia
1	Funcionabilidad	0.9	90%
2	Fiabilidad	0.99	99%
3	Usabilidad	0.91	91%
4	Mantenibilidad	1	100%
5	Eficiencia	91	91%
6	Portabilidad	83	83%
Evaluación de la Calidad Global		0.92	92%

Fuente: [Elaboración propia en base a ISO 9126].

Según (Pressman; 2010), se define lo siguiente:

- Un nivel de aceptación satisfactorio, indica que los valores de la evaluación global se encuentran en el rango de 60-100.

- Un nivel de aceptación marginal, indica que los valores de la evaluación global se encuentran en el rango de 40-60.
- Un nivel de aceptación insatisfactorio, indica que los valores de la evaluación global se encuentran en el rango de 0-40.
- Por lo tanto la referencia global de calidad es de 92%

Observando el porcentaje, el trabajo realizado está en un nivel satisfactorio.

4.2 Estimación de Costos de Software

Existen distintos métodos para la estimación de costos de desarrollo de software, estos métodos no son otra cosa que establecer una relación matemática entre el esfuerzo y el tiempo de desarrollo.

4.2.1 Teoría de costos

El Modelo Constructivo de Costes (Constructive Cost Model) fue desarrollado por B. W. Boehm a finales de los 70 y comienzos de los 80, exponiéndolo detalladamente en su libro "Software Engineering Economics" [Prentice-Hall, 1981].

COCOMO es una jerarquía de modelos de estimación de costes software que incluye sub-modelos *básico*, *intermedio* y *detallado*.

Las ecuaciones de estimación del esfuerzo de desarrollo tienen la forma:

- S el número de miles de líneas de código fuente.
- M(x) es un multiplicador que depende de 15 atributos.

En la tabla X se muestra los coeficientes para los diferentes modos

Tabla 4.4 Coeficientes para los modos.

	Básico		Intermedio	
Modo	Ai	Bi	Ai	Bi
Organico	2.4	1.05	3.2	1.05

Semi- encajado	3.0	1.12	3.0	1.12
Enpotrado	3.6	1.2	2.8	1.2

Fuente: [Roger Pressman, 2020].

Se utilizará para la tesis, el modelo básico de COCOMO, por ser el indicado para un programa de nivel de complejidad media.

En la siguiente tabla se especifican los módulos que serán desarrollados en el sistema con su respectiva cantidad de líneas de código.

Tabla 4.5 Tabla Costo del sistema según líneas de código.

Modulos	Cantidad de líneas de Código/LDC
Inicio de Sesión	
login	173
Reconocimiento	
reconocer_x	596
Gestion de usuario	
lista_usuario	42
Docente	
Nuevo_reg	578
perfilDocente	364
Entrenamiento	520
funciones	
funcion_listar_reg_ent_sal_por_fecha	48
funcion_listar_reg_ent_sal	53
funcion_centrar	13
funcion_listar_entrenamiento	31
funcion_listar_nuevo_registro	46
funcion_listar_usuario	48
TOTAL	2512

Fuente: [Elaboración propia, 2020].

Teniendo en cuenta que el sistema tiene dos mil quinientas doce líneas de código fuente, aproximada para la obtención del parámetro esfuerzo total en meses de los programadores necesarios para el desarrollo del sistema.

El modelo consiste en matrices orientadas al tamaño tratan de establecer la productividad, el costo y la funcionalidad en base a las líneas de código de programa expresados en KLDC.

Esfuerzo de la Tesis:

$$k_m = 2.4(S_k)^{1.05}$$

Donde:

$$S_k = 2512 \text{ LDC} = 2.512 \text{ KLDC}$$

Reemplazando en la fórmula, se tiene:

$$k_m = 2.4(S_k)^{1.05}$$

$$k_m = 2.4(2.512)^{1.05}$$

$$k_m = 2.4(2.512)$$

$$k_m = 6.0288$$

El resultado obtenido significa que el esfuerzo requerido para esta tesis es de 6 personas mes.

Tiempo de Desarrollo o Duración del Proyecto:

$$T_d = 2.5(k_m)^{0.38}$$

Donde $k_m = 2.05$ entonces por la fórmula se tiene:

$$T_d = 2.5(k_m)^{0.38}$$

$$T_d = 2.5(6.0288)^{0.38}$$

$$T_d = 2.5(1.98)$$

$$T_d = \text{tiempo requerido} = 4.95 \text{ meses}$$

El resultado obtenido significa que el tiempo requerido para esta tesis es de 4.95 meses.

Número de Personas para la Tesis mes:

$$\frac{k_m}{T_d}$$

Conociendo los datos $k_m = 6.0288$, $T_d = 4.95$ meses, finalmente se tiene:

$$\frac{k_m}{T_d} = \frac{6.0288}{4.95}$$

$$\frac{k_m}{T_d} = N^{\circ} \text{ de personas mes} = 1.22$$

Entonces para la tesis propuesta es necesario el esfuerzo de 1.22 personas por mes.

En la tabla se observa algunos parámetros que se utilizaron para los cálculos en las diferentes etapas.

Por lo tanto, si el sistema contiene tantas líneas LDC es decir:

$$2.512 \text{ KLDC} = S_k$$

Se estableció el sueldo del Programador SP=5 \$us/hora.

- **Etapas de Análisis e investigación preliminar**

Se tienen 4 semanas para esta etapa, tomando en cuenta el Sueldo del Programador SP=5 \$us / hora, se tomó 120 horas hábiles, por lo tanto:

$$EA = 160 \text{ horas} * 5 \text{ $us / hora} = 800 \text{ $us}$$

- **Etapas de Diseño del sistema**

Se tienen 2 semanas para esta etapa

$$EDS = 160 \text{ horas} * 5 \text{ $us / hora} = 800 \text{ $us}$$

- **Etapa de Desarrollo o Codificación**

Se tiene 8 semanas para esta etapa

$$ED = 320 \text{ horas} * 5 \text{ \$us/hora} = 1600 \text{ \$us}$$

- **Etapa de implementación de Pruebas**

Se tiene 4 semanas para esta etapa.

$$EIP = 160 \text{ horas} * 5 \text{ \$us / hora} = 800 \text{ \$us}$$

En la tabla se observa el costo total de las etapas de análisis e investigación preliminar, diseño del sistema, desarrollo o codificación e implementación y pruebas.

Tabla 4.6 Costo total etapas del Software.

Etapa	Costo en \$us
Análisis e investigación preliminar	800
Diseño del sistema	800
Desarrollo o Codificación	1600
Implementación de Pruebas	800
Total	4000

Fuente: [Elaboración propia, 2020].

Se obtiene el costo por etapas de la tesis 4000 \$us, después de realizarse la debida planificación de cada una de estas etapas.

Estimación de recursos

a) Hardware

Los recursos necesarios en cuanto a hardware se refieren se detallan en la tabla

Tabla 4.7 Recursos de Hardware.

Hardware
Ordenadores
Servidor
Unidades respaldo
Otros

Fuente: [Elaboración propia, 2020].

b) Software

En la tabla se tiene el detalle de las versiones de software a usar conjuntamente con los precios de sus licencias.

Tabla 4.8 Recursos de Software de desarrollo.

Nombre	Versión	
Microsoft Windows	Win10	100
Office (Excel)	2010	28
MatLab	R2018b	940
XAMPP	7.3.12	Libre
MySql	8.0.20	Libre
Total		1068

Fuente: [Paginas Microsoft y Software Libre].

c) Materiales

En la tabla se tiene el detalle del costo de insumos a utilizar en la tesis

Tabla 4.9 De Insumos.

Materiales	Costo en \$us
CD, DVD, Material de Escritorio	50
Total	50

Fuente: [Elaboración propia, 2020].

Costo total del Proyecto

De acuerdo a los datos obtenidos el costo del desarrollo del sistema es de 4000 \$us., monto al cual debe añadirse los costos de licencia de software, material de escritorio y otros.

Tabla 4.10 Costo total del Proyecto.

Concepto	Costo en \$us
Costo análisis, diseño, desarrollo e implementación	4000
Costos de licencia de software	1068
Costos de insumos	50
Total	5118

Fuente: [Elaboración propia, 2020].

Por lo tanto, el desarrollo del sistema tendrá un costo total de 5118 Dólares Americanos cambiando a bolivianos.

Entonces:

$$1 \text{ \$us} = 6.85$$

Tenemos: $5118 \text{ \$us} = 35058.3 \text{ bolivianos.}$

5. PRUEBAS DE HIPÓTESIS

5.1 Prueba de Hipótesis T

Una prueba t es una prueba de hipótesis de la media de una o dos poblaciones distribuidas normalmente. Aunque existen varios tipos de prueba t para situaciones diferentes, en todas se utiliza un estadístico de prueba que sigue una distribución t bajo la hipótesis nula:

Se usa las distancias de una cara para la muestra 1, y de la misma cara se toma otras distancias para la muestra 2.

- Se plantea las hipótesis:

Hipótesis alternativa (H_1)

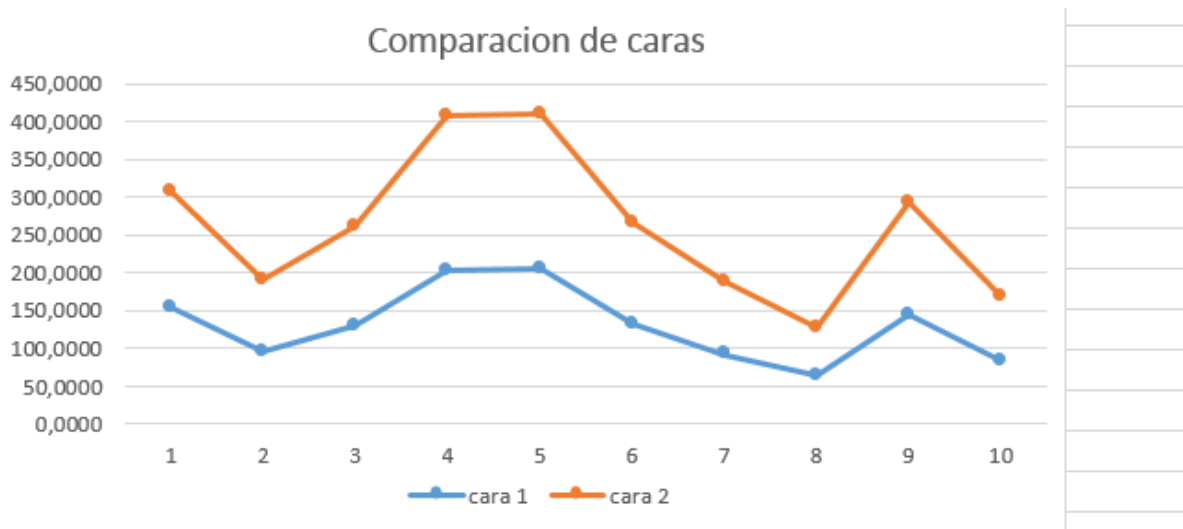
Las muestras de la cara 1 son similares a las muestras de la cara 2, por lo cual los datos planteados son factibles para el reconocimiento facial.

Figura 5.1 Datos de entrada y salida de la hipótesis T.

cara 1	cara 2	Prueba t para medias de dos muestras emparejadas		
153,8830	153,8830			
96,3872	95,5708		<i>Variable 1</i>	<i>Variable 2</i>
130,0780	131,9410	Media	130,51841	131,65386
203,4760	203,3680	Varianza	2339,34181	2280,51298
204,8220	205,5890	Observaciones	10	10
132,5240	133,6860	Coefficiente de correlación de Pearson	0,99968647	
92,2239	95,1972	Diferencia hipotética de las medias	0	
63,5059	64,7237	Grados de libertad	9	
145,7780	146,5730	Estadístico t	-2,6594112	
82,5061	86,0069	P(T<=t) una cola	0,01303547	
		Valor crítico de t (una cola)	1,83311293	
		P(T<=t) dos colas	0,02607094	
		Valor crítico de t (dos colas)	2,26215716	

Fuente: [Elaboración propia, 2020].

Figura 5.2 Comparación de caras.



Fuente: [Elaboración propia, 2020].

$P (0.01303547 \leq 0.05)$ por lo tanto se llega a la conclusión que se acepta la Hipótesis alternativa y se rechaza la Hipótesis nula

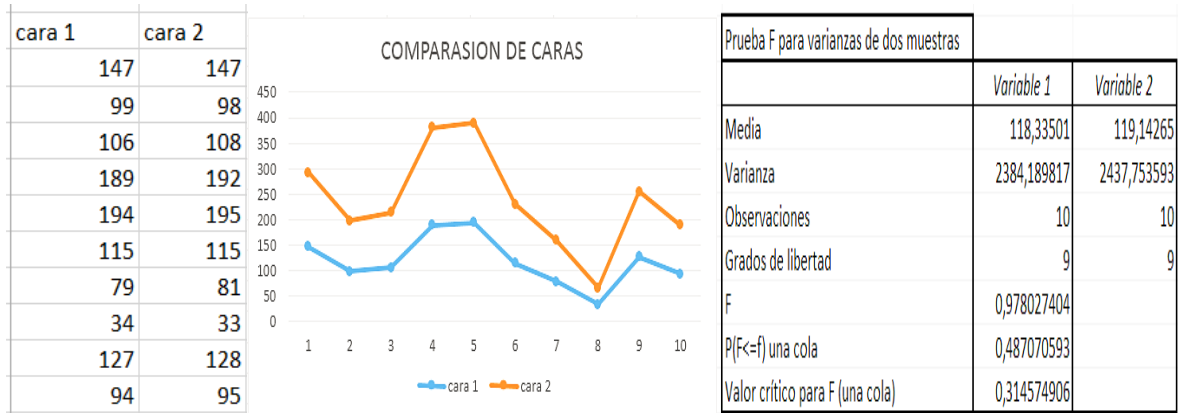
5.2 Prueba de hipótesis F

A diferencia de otras pruebas de medias que se basan en la diferencia existente entre dos valores, el análisis de varianza emplea la razón de las estimaciones, dividiendo la estimación intermedia entre la estimación interna.

Figura 5.3 Datos de entrada y s.

$$Razón F = \frac{s_x^2}{s_w^2} = \frac{ns_{\bar{x}}^2}{(s_1^2 + s_2^2 + s_3^2 + \dots + s_k^2)/k} \quad (2.20)$$

Figura 5.4 Salida de la hipótesis F.



Fuente: [Elaboración propia, 2020].

Hipótesis nula (Ho)

Se rechaza la hipótesis nula, por el intervalo central se encuentra con 0,95, por el tanto se acepta la hipótesis alterna.

6. CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

El utilizado en la presente tesis, es la tecnología de las redes neuronales artificiales, en específico las Redes de Función Base Radial (RBF). Utilizando el método de las distancias de puntos característicos, como ser los ojos, nariz y boca. Esas distancias son utilizadas para entrenar a la red neuronal artificial (RBF), y así poder reconocer a la persona, cada que ingrese su rostro al sistema, y los datos obtenidos son guardados automáticamente en una base de datos, para su posterior consulta las veces que sea requerida.

6.2 Recomendaciones

Para continuar mejorando el sistema y el tema de esta investigación se recomienda:

- Aumentar el número de rasgos y puntos característicos en los rostros de las personas para realizar una distinción más profunda y así poder entrenar mejor la red neuronal (RBF).
- Dotar al sistema de seguridad, contra comportamientos irregulares que tiene el usuario para burlar el sistema.

6.3 Bibliografía

De abajo, N. Gómez, A. (2001). Introducción a la inteligencia artificial: sistemas expertos, redes neuronales artificiales y computación evolutiva, Universidad de Oviedo.

- Carlos T. Marcos P. Jaime R. (2011) Sistemas Biométricos Universidad de las Palmas de Gran Canaria

- Córdova, J (2010) Introducción a MATLAB para Ingenieros

- De Abajo, N. Gómez, A. (2001). Introducción a la Inteligencia Artificial: Sistemas Expertos, Redes Neuronales Artificiales y Computación Evolutiva, Universidad de Oviedo.

- Enrique C. (2004) Técnicas de Reconocimiento Facial Mediante Redes Neuronales Departamento de Tecnología Fotónica Facultad de Informática, Madrid.

- FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA (Quito) “Sistema de Control de Acceso Basado en el Perfil Lateral de una Persona Utilizando Visión Artificial” Juan Pablo Rodríguez Salvatierra (octubre 2009).

- Gilat, a Matlab una Introduction con Ejemplos, Department of Mechanical Engineering, The Ohio State University

- González, M Sistemas Conexionistas (Redes de Base Radial)

- Introducción al Procesamiento Digital de imágenes con Matlab
[URL:http://www.unioviedo.es/compnum/laboratorios_web/laborat03_intro_imagen/laborat03.html](http://www.unioviedo.es/compnum/laboratorios_web/laborat03_intro_imagen/laborat03.html) [consulta: 18 de Mayo de 2015]

- López, O. (2012) Sistema de Visión Artificial. Instituto tecnológico de Hermosillo.

- Paz, J (2009) Procesamiento de Imágenes con Matlab España

 - Politécnica de Madrid-UPM. Dr. Diego Andina de la Fuente. M. en I. Antonio Vega Corona de la Facultad de Ingeniería Mecánica, Redes Neuronales Artificiales
URL: <http://www.gc.ssr.upm.es/inves/neural/ann2/anntutorial.html>
[Consulta: 25 de Mayo de 2014]

 - Procesamiento de Imágenes con MatLab. 2014 Domingo Mery.
URL:<http://dmery.ing.puc.cl/index.php/teaching/imagenes/matlab/>
[Consulta: 30 de Mayo de 2014]

 - Ramírez Q. Juan A. Chacón M. Redes Neuronales Artificiales para el Proceso de Imágenes, una Versión de la Última Década REVISTA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y COMPUTACIÓN, Vol.9 No.1, JULIO 2011.
- Diseño de un sistema de reconocimiento facial (Serrano, Soria, Martin, 2010)
- ¿Que son las redes neuronales?
URL:http://info.fisica.uson.mx/arnulfo.castellanos/archivos_html/quesonredneu.htm
[Consulta: 26 de Mayo de 2014]

6.4 Webgrafía

<http://ocwus.us.es/matematica-aplicada>

<http://camarasdeseguridad.over-blog.net/pages/Accesos-biometricos-definicion-5930945.html>

<http://www.monografias.com/trabajos57/huellas-lofoscopicas/huellas-lofoscopicas3.shtml>

<http://www.muyinteresante.es/tecnologia/articulo/biometria-tu-cara-el-password-mas-seguro-501449137231>

<http://es.slideshare.net/estebansaavedra/biometria-y-patrones-para-la-identificacion-humana>

<http://www.comunicacampeche.com.mx/Php/actualidad.php?id=141755>

<http://joel-redesneuronalesartificiales.blogspot.com/>

<http://www.adrformacion.com/cursos/flashcs5/leccion2/tutorial19.html>

<http://pt.slideshare.net/mariagraciao/diagnostico-en-ortodoncia-caso-clinico>

[https://repository.eafit.edu.co/xmlui/bitstream/handle/10784/4617/34 Metodologia_reconocimiento_emociones.pdf?sequence=1&isAllowed=y](https://repository.eafit.edu.co/xmlui/bitstream/handle/10784/4617/34_Metodologia_reconocimiento_emociones.pdf?sequence=1&isAllowed=y)

