



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Sistemas

**Sistema inteligente basado en wavelets
bidimensionales y máquinas de vectores de soporte
para el reconocimiento automático de emociones
faciales**

TESIS

Para optar el Título Profesional de Ingeniera de Sistemas

AUTOR

Fiorella Andrea ALEJOS YARASCA

ASESOR

Santiago Domingo MOQUILLAZA HENRÍQUEZ

Lima, Perú

2021



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Alejos, Y. (2021). *Sistema inteligente basado en wavelets bidimensionales y máquinas de vectores de soporte para el reconocimiento automático de emociones faciales*. [Tesis de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Sistemas]. Repositorio institucional Cybertesis UNMSM.

HOJA DE METADATOS COMPLEMENTARIOS

Código ORCID del autor	—
DNI o pasaporte del autor	76294277
Código ORCID del asesor	https://orcid.org/0000-0001-9531-881X
DNI o pasaporte del asesor	08280889
Grupo de investigación	INNOVANDO SISTEMAS INTELIGENTES
Agencia financiadora	NO
Ubicación geográfica donde se desarrolló la investigación	Perú, Lima, Lima, Carabayllo Latitud: -11.794993 Longitud: -76.989292
Año ó rango de años en que se realizó la investigación	2018-2021
Disciplinas OCDE	Ingeniería de sistemas y comunicaciones https://purl.org/pe-repo/ocde/ford#2.02.04



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA
Escuela Profesional de Ingeniería de Sistemas

Acta Virtual de Sustentación de Tesis

Siendo las 16:20 horas del día 20 de mayo del año 2021, se reunieron virtualmente los docentes designados como miembros de Jurado de Tesis, presidido por la Dra. Nora Bertha La Serna Palomino, Mg. Rolando Maguiña Pérez (Miembro) y el Mg. Santiago Domingo Moquillaza Henríquez (Miembro Asesor), usando la plataforma Meet (meet.google.com/dqr-kpye-eeo), para la sustentación Virtual de la tesis Intitulada:

“Sistema inteligente basado en wavelets bidimensionales y máquina de vectores de soporte para el reconocimiento automático de emociones faciales”, de la Bachiller: **Fiorella Andrea Alejos Yarasca**; para obtener el Título Profesional de Ingeniero de Sistemas.

Acto seguido de la exposición de la Tesis, el Presidente invitó a la Bachiller a responder las preguntas formuladas por los Miembros del Jurado.

La Bachiller, en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez a las preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros del Jurado, la bachiller obtuvo la nota de **18 (dieciocho)**

A continuación la Presidenta del Jurado Dra. Nora Bertha La Serna Palomino, declara a la Bachiller **Ingeniero de Sistemas**.

Siendo las 17:15 horas, se levantó la sesión.



Firmado digitalmente por LA SERNA
PALOMINO Nora Bertha FAU
20148092282 soft
Motivo: Soy el autor del documento
Fecha: 31.05.2021 02:21:03 -05:00

Dra. Nora Bertha La Serna Palomino
Presidente

Mg. Rolando Maguiña Pérez
Miembro

Mg. Santiago D. Moquillaza Henríquez
Miembro Asesor

FICHA CATALOGRÁFICA

ALEJOS YARASCA, FIORELLA ANDREA

SISTEMA INTELIGENTE BASADO EN WAVELETS BIDIMENSIONALES Y
MÁQUINAS DE VECTORES DE SOPORTE PARA EL RECONOCIMIENTO
AUTOMÁTICO DE EMOCIONES FACIALES.

Clasificación de Imágenes

Lima, Perú 2021

Tesis, Facultad de Ingeniería de Sistemas, Escuela Profesional Ingeniería de Sistemas,
Post grado, Universidad Nacional Mayor de San Marcos

Formato 28x20 cm

Páginas 98 páginas

Dedicatoria

Este trabajo está dedicado a Dios, mi familia, amigos, profesores y colegas por brindarme orientación, confianza, tiempo y afecto en estos años de mi vida.

Agradecimientos

Agradezco a mis asesores: Santiago Moquillaza, Nora La Serna y Cesar Luza por su apoyo, orientación y aporte de su valiosa experiencia a lo largo del desarrollo de este importante proyecto de tesis. También, agradecer la colaboración de Walter Llana durante el desarrollo de este trabajo.

Agradezco asimismo a la Facultad de Ingeniería de Sistemas e Informática.

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

**SISTEMA INTELIGENTE BASADO EN WAVELETS BIDIMENSIONALES Y
MÁQUINAS DE VECTORES DE SOPORTE PARA EL RECONOCIMIENTO
AUTOMÁTICO DE EMOCIONES FACIALES**

Autor : ALEJOS YARASCA, Fiorella Andrea
Asesor : MOQUILLAZA HENRÍQUEZ, Santiago Domingo
Título : Tesis, para optar el Título Profesional de Ingeniero de Sistemas
Fecha : Febrero, 2021

Resumen

El reconocimiento de emociones faciales es muy importante para la comunicación social. Por lo cual, a lo largo de los años se han realizado numerosos estudios e investigaciones sobre el reconocimiento automático de emociones. Generalmente, los sistemas de reconocimiento de emociones faciales se componen de las fases de preprocesamiento de la imagen, extracción de características y clasificación de la expresión facial. La presente tesis propone un método para el reconocimiento automático de emociones faciales en imágenes digitales. Este método utiliza ecualización de histogramas para mejorar las condiciones de iluminación como parte del preprocesamiento de imágenes, transformada de wavelet discreta bidimensional y algoritmo PCA para la extracción y reducción de características respectivamente. Finalmente se utiliza un clasificador SVM lineal para predecir la emoción. Los experimentos se realizaron usando la base de datos JAFFE. El sistema obtuvo una precisión promedio mayor al 93%. Además reconoció con mayor precisión las emociones: Felicidad, sorpresa y neutral; con lo cual se demostró la efectividad del sistema propuesto.

Palabras claves: Reconocimiento de emociones faciales, clasificación de expresiones faciales, extracción de características, pre-procesamiento de imágenes

MAJOR UNIVERSITY OF SAN MARCOS
FACULTY OF SYSTEMS ENGINEERING
PROFESSIONAL SCHOOL OF SYSTEMS ENGINEERING

**INTELLIGENT SYSTEM BASED ON TWO-DIMENSIONAL WAVELETS AND
SUPPORT VECTOR MACHINES FOR THE AUTOMATIC RECOGNITION
OF FACIAL EMOTIONS**

Author : ALEJOS YARASCA, Fiorella Andrea
Adviser : MOQUILLAZA HENRÍQUEZ, Santiago Domingo
Title : Thesis, to choose the Professional Title System Engineer
Date : February 2021

Abstract

Facial emotion recognition is very important for social communication. Whereby through the years it has done many studies and researches about automatic emotion recognition. Generally, the facial emotion recognition systems are composed of the pre-processing phase, the features extraction phase and the classification phase.

This thesis proposes a method for automatic facial emotion recognition in digital images. This method uses histogram equalization to improve the light conditions in the pre-processing of images, bi dimensional discrete wavelet transform and PCA algorithm are used to extract and reduce features. Finally it uses a linear support vector machine sorter to predict emotions.

The experiments were made using the JAFFE database. The system achieves more than 93% in average accuracy. Also recognizes better the following emotions: Happiness, surprise and neutral. Whereby it proved its effectiveness.

Key words: Facial emotion recognition, facial expression classification, feature extraction, image pre-processing

Índice

Dedicatoria	6
Agradecimientos	7
Resumen.....	8
Abstract	9
Índice.....	11
Lista de Figuras	14
Lista de Tablas	16
Introducción	17
Capítulo I Planteamiento metodológico.....	18
1.1 Antecedentes	18
1.2 Definición o formulación del problema	19
1.2.1 Problema principal	19
1.2.2 Problemas específicos	19
1.3 Objetivos	20
1.3.1 Objetivo General	20
1.3.2 Objetivos Específicos	20
1.4 Justificación	20
1.4.1 Justificación teórica	20
1.4.2 Justificación práctica	21
1.5 Alcance	21
1.6 Organización de la tesis.....	22
Capítulo II Marco teórico	23
2.1 Emoción.....	23
2.2 Expresión facial	23
2.3 Emociones básicas.....	24
2.4 Reconocimiento de emociones faciales	24
2.5 Fases del reconocimiento de emociones automático.....	24
2.6 Preprocesamiento de imágenes.....	25
2.7 Extracción de características	25
2.8 Clasificación de expresiones faciales.....	25
2.9 Base de datos de expresiones faciales	26
2.10 Red Neuronal Convolutacional (CNN)	26

2.11 Base de datos de expresión facial femenina japonesa (JAFPE)	27
2.12 Base de datos Cohn-Kanade (CK)	29
2.13 Base de datos Extended Cohn–Kanade (CK+)	29
2.14 Base de datos FER-2013	30
2.15 Base de datos Real-world Affective Faces (RAF-DB)	30
2.16 Ecualización de histograma	31
2.17 Transformada de wavelet discreta (DWT)	34
2.18 Transformada discreta de wavelet bidimensional (2D-DWT)	35
2.19 Análisis de componentes principales (PCA)	37
2.20 Máquinas de vectores de soporte o Máquinas de soporte vectorial	40
Capítulo III Estado del arte metodológico	43
3.1 Artículos	43
IV Aporte teórico	50
4.1 Comparación de las tecnologías.....	50
4.2 Estrategia para el diseño de la solución y selección de tecnologías	56
4.3 Metodología	57
4.3.1 Base de datos de expresiones faciales	58
4.3.2 Preprocesamiento de imágenes.....	58
4.3.3 Extracción de características	58
4.3.4 Clasificación.....	59
4.3.5 Entrenamiento del reconocedor	59
4.3.6 Pruebas.....	59
4.3.7 Validación del sistema	59
4.4 Tecnologías a utilizar	61
4.4.1 Hardware.....	61
4.4.2 Software	61
Capítulo V Implementación	65
5.1 Desarrollo de la solución	65
5.1.1 Preparación de la base de datos de expresiones faciales	65
5.1.2 Preparación del ambiente de desarrollo.....	66
5.1.3 Componentes software	69
5.1.4 Pre-procesamiento de imágenes	70
5.1.5 Extracción de características y reducción de dimensionalidad usando wavelets bidimensionales y PCA:	77

5.1.6 Configuración del clasificador SVM y entrenamiento:.....	79
5.1.7 Evaluación	81
5.2 Resultados experimentales	81
5.3 Comparación con otros métodos.....	88
Capítulo VI Discusión, Conclusiones y recomendaciones	92
6.1 Discusión	92
6.2 Conclusiones y recomendaciones	93
REFERENCIAS BIBLIOGRÁFICAS	94

Lista de Figuras

Figura 1: Ejemplo de expresiones faciales en las imágenes de la base de datos CK+	24
Figura 2: Ejemplo de de clasificación de expresiones faciales	26
Figura 3: Estructura general de una implementación de Red Neuronal Convolutiva	27
Figura 4: Imágenes de base de datos JAFFE.....	28
Figura 5: Aparato con el cual se fotografió las expresiones faciales en JAFFE	28
Figura 6: Imágenes de la base de datos CK+	30
Figura 7: Proceso de construcción de la base de datos RAF-DB	31
Figura 8: Ejemplo de imágenes de las seis emociones básicas y las 20 clases de emociones de la base de datos RAF-DB.....	31
Figura 9: Imagen tomada de la base de datos JAFFE leída usando biblioteca OpenCV	33
Figura 10: Histograma de la imagen original tomada de JAFFE elaborado usando Python.....	33
Figura 11: Imagen después de aplicar ecualización de histograma a una imagen tomada de la base de datos JAFFE, elaborado usando Python.....	34
Figura 12: Histograma de la nueva imagen obtenida después de aplicar la técnica ecualización de histograma, elaborado usando Python.....	34
Figura 13: Proceso de descomposición de una imagen usando DWT.....	35
Figura 14: a) Nivel uno de descomposición wavelet b) Nivel dos de descomposición wavelet.	36
Figura 15: Coeficientes LL2, LH2, HL2 y HH2 de una imagen después de aplicar 2D-DWT a la imagen original, figura elaborada usando Python	37
Figura 16: Conjunto de datos Iris proyectado usando PCA	40
Figura 17: Proceso de transformación – Máquinas de Vectores de Soporte	41
Figura 18: Estructura del modelo propuesto	44
Figura 19: Diagrama de flujo del marco de representación de la expresión facial multimodal propuesto.....	45
Figura 20: Procedimiento en la fase de entrenamiento	47
Figura 21: Flujo de trabajo del modelo propuesto	48
Figura 22: Proceso de extracción de características de Gabor	49
Figura 23: Arquitectura del sistema de reconocimiento de expresiones faciales propuesto	57
Figura 24: Logo python.....	62
Figura 25: Logo Jupyter Notebook	62
Figura 26: Logo OpenCV.....	63
Figura 27: Logo Numpy.....	63
Figura 28: Logo Scikit learn.....	63
Figura 29: Logo Matplotlib.....	64
Figura 30: Logo Dlib.....	64
Figura 31: Logo Pandas.....	64
Figura 32: Imagen de la carpeta RECONOCEDOR donde se colocó las imágenes descargadas de la base de datos JAFFE.....	66
Figura 33: Componentes software.....	70
Figura 34: Imagen de la base de datos JAFFE leída usando OpenCV en Python.....	71
Figura 35: Puntos de referencia identificados en el rostro	72
Figura 36: Detección de ojos en la imagen	73
Figura 37: Localización de los puntos centrales de los ojos en el rostro	73

Figura 38: Rotación de la imagen	74
Figura 39: Localización del punto medio entre los ojos en la imagen	75
Figura 40: Región de interés (ROI)	75
Figura 41: Imagen redimensionada	76
Figura 42: Imagen después de ecualizar su histograma	76
Figura 43: Coeficientes LL2 y LH2	77
Figura 44: Coeficientes HL2 y HH2	78
Figura 45: Datos proyectados en de dos componentes utilizando PCA	79
Figura 46: Resultados de la clasificación de emociones	82
Figura 47: Matriz de confusión para una clasificación binaria	83
Figura 48: Matriz de confusión sin normalización, elaborado en Python	84
Figura 49: Matriz de confusión normalizada, elaborado en Python	85
Figura 50: Arquitectura del CNN implementado, basado en (Analytics Vidhya, s.f.)	89
Figura 51: Resultados usando CNN sobre el conjunto de validación	90
Figura 52: Línea de aprendizaje de pérdida y precisión	90

Lista de Tablas

Tabla 1: Artículos analizados	50
Tabla 2: Comparación de ventajas, desventajas y métodos/técnicas utilizadas	53
Tabla 3: Criterios de evaluación.....	54
Tabla 4: Pesos de los criterios	54
Tabla 5: Evaluación de métodos	55
Tabla 6: Representación de clases.....	71
Tabla 7: Parámetros del clasificador SVM.....	80
Tabla 8: Matriz de confusión	84
Tabla 9: Matriz de confusión normalizada.....	85
Tabla 10: Reporte de clasificación	86
Tabla 11: Tiempos obtenidos por etapas del método propuesto	87
Tabla 12: Comparación de precisiones en JAFFE y CK+.....	87
Tabla 13: Comparación entre métodos.....	88
Tabla 14: Comparación de precisión después de variar el clasificador.....	91

Introducción

Los seres humanos comunican emociones mediante expresiones faciales, los músculos de la cara intervienen para formar expresiones faciales particulares.

El reconocimiento facial de la expresión emocional es la capacidad de todos los individuos de reconocer formas básicas de expresión afectiva, las cuales aparecen en los rostros de las personas (Iglesias, Del Castillo, & Muñoz, 2016). La capacidad de reconocer la expresión facial emocional de los demás es la base de gran parte de la interacción social exitosa (Connolly, Lefevre, Young, & Lewis, 2019).

A lo largo de los años se han realizado numerosos estudios y experimentos en torno al reconocimiento de emociones y su universalidad. Adelantado a su época, Charles Darwin realizó experimentos de reconocimiento de emociones con personas cercanas a él utilizando fotografías les preguntaba por la emoción observada. Además postuló que seres humanos de diferentes culturas incluso animales realizan los mismos movimientos emocionales (Darwin, 1872). Años más tarde el psicólogo Paul Ekman realizó un experimento en una tribu de Papúa Guinea donde comprobó que ciertas emociones pueden ser encontradas en diferentes culturas, así mismo propuso 6 emociones básicas: Ira, asco, miedo, alegría, tristeza y sorpresa (Ekman, Sorenson, & Friesen, Pan-cultural elements in facial displays of emotion, 1969).

En los últimos años gracias al avance tecnológico se han desarrollado sistemas inteligentes para el reconocimiento automático de emociones, estos sistemas son muy beneficiosos en la vida humana por su aplicación en: Publicidad, marketing, seguridad y salud. Generalmente los sistemas de reconocimiento automático de emociones faciales se componen de tres pasos: Detección de rostro, extracción de características y clasificación de características.

Es por ello que el presente trabajo de tesis propone desarrollar un sistema para el reconocimiento automático de emociones faciales en imágenes digitales basado en Wavelets con la finalidad de brindar solución efectiva al problema del reconocimiento de emociones faciales.

Capítulo I Planteamiento metodológico

1.1 Antecedentes

El reconocimiento de emociones humanas es un tema complejo y demanda colaboración de numerosos campos y disciplinas. En las últimas décadas se han desarrollado diversos estudios y experimentos entorno al problema de clasificación de expresiones faciales, los cuales serán detallados a continuación:

Como primer antecedente se presenta el estudio del famoso naturalista Charles Darwin quien postuló que tanto los seres humanos como los animales cercanos al hombre, por ejemplo primates y perros, muestran sus estados emocionales a través de los mismos movimientos musculares. También desarrolló experimentos simples que consistían en mostrar a sus visitantes un conjunto de fotografías de rostros humanos, algunas de ellos mostraban los músculos artificialmente contraídos por sondas eléctricas, y preguntarles por la emoción que reflejaban usando sus propias palabras para luego documentar los resultados en tablas (Darwin, 1872). Estas tablas en manuscritos se encuentran actualmente en la biblioteca digital de la universidad de Cambridge.

Años más tarde en 1969 el psicólogo Paul Ekman dirigió un experimento en tribus aisladas de Papúa Nueva Guinea. El experimento consistió en mostrarle a cada miembro de la tribu fotografías de rostros humanos pertenecientes a otras culturas diferentes con las cuales no habían tenido alguna interacción previa y pedirles que las relacionen con enunciados diseñados para desencadenar las emociones observadas en las fotografías. Se obtuvo reconocimientos similares de felicidad, enojo y miedo en todas las muestras, y disgusto y sorpresa en dos de las tres muestras. Este estudio demostró que existen elementos pan culturales en la forma en que se expresan las emociones humanas (Ekman, Sorenson, & Friesen, Pan-cultural elements in facial displays of emotion, 1969).

Ekman en colaboración de Friesen y Hager desarrollaron el Sistema de Codificación de Acción Facial (FACS). Este sistema consistió en una guía visual donde se asociaba un conjunto de movimientos musculares faciales a una emoción (Ekman, Friesen, & Hager, Facial Action Coding System - The Manual on CD-ROM [CD-ROM], 2002).

En el año 2009 Rana el Kaliouby y Rosalind Picard fundaron la compañía Affective la cual desarrolló tecnología como Affdex, software firma de la compañía el cual busca una cara; si hay varias caras, aísla cada una. Luego identifica las regiones principales de la cara (boca, nariz, ojos, cejas) para asignar puntos verdes a cada una, también busca la textura cambiante de la piel (Distribución de arrugas y surcos en el rostro). Después combina la información de los puntos verdes con la textura del rostro para construir modelos detallados de la cara. Este algoritmo identifica una expresión emocional comparándola con otras numerosas expresiones que ha analizado previamente. Actualmente sus soluciones ayudan a las marcas a mejorar sus mensajes publicitarios y de marketing (Khatchadourian, 2015).

1.2 Definición o formulación del problema

1.2.1 Problema principal

En el Perú actualmente es incipiente el desarrollo de sistemas notables de reconocimiento automático de emociones faciales a pesar de tratarse de una industria en expansión y de ser de una necesidad actual; dado que se necesita saber el estado emocional de la persona, lo cual permite ayudar al ser humano ya sea en el trabajo y en el estudio.

De manera que podemos plantearnos el problema:

- ¿De qué manera se puede reconocer automáticamente las emociones faciales humanas?

1.2.2 Problemas específicos

- ¿Con qué precisión el sistema identificará la emoción en un rostro digitalizado?
- ¿Cuáles son las emociones a ser reconocidas?

1.3 Objetivos

1.3.1 Objetivo General

- Desarrollar un Sistema inteligente basado en Wavelets Bidimensionales y Máquinas de Vectores de Soporte Lineal para el reconocimiento automático de emociones faciales.

1.3.2 Objetivos Específicos

- Identificar emociones en un grupo de personas a partir de sus rostros digitalizados, con una precisión de al menos 80%.
- Identificar, a partir del presente estudio, qué y cuántas emociones serán reconocidas por el sistema inteligente.

1.4 Justificación

1.4.1 Justificación teórica

El presente trabajo de tesis se realiza con el propósito de aportar al conocimiento existente en torno al reconocimiento automático de emociones faciales, como instrumento de evaluación de la precisión obtenida al utilizar Wavelets bidimensional y Máquinas de Vectores de Soporte en el desarrollo de un sistema de clasificación de la expresión del rostro; cuyos resultados podrán ser utilizados como referencia o antecedente en el futuro por investigadores que se encuentren en la búsqueda de técnicas y métodos de inteligencia artificial para la construcción de sus propios sistemas inteligentes.

1.4.2 Justificación práctica

Esta investigación se realiza porque existe la necesidad de conocer la emoción desencadenada en los consumidores antes publicidad y marketing de servicios y/o productos. Los proveedores de servicios que presten atención a los factores afectivos y construyan “relaciones” con sus clientes tendrán una ventaja competitiva en el mercado futuro(Erevelles, Shuba, & Rangel, 2003).

Medir la satisfacción del cliente proporciona una indicación del éxito de una empresa y las técnicas de computación afectiva proporcionan un nuevo horizonte en el análisis cuantitativo del estado emocional del consumidor usando métodos de inteligencia artificial(Ren & Quan, 2012).

1.5 Alcance

En el presente proyecto de tesis se pretende desarrollar un sistema de reconocimiento automático de emociones faciales utilizando wavelets bidimensionales para la extracción de características.

Para la clasificación de características se utilizarán Máquinas de Vectores de Soporte lineal (SVM).

El sistema tendrá como entrada imágenes digitales y como salida se obtendrá la expresión facial reconocida.

El sistema reconocerá las siguientes emociones humanas: Felicidad, enojo, desagrado, miedo, tristeza, sorpresa y neutral.

Dado que el Perú no cuenta con una base de datos de expresiones faciales, se utilizarán imágenes de una base de datos de expresiones faciales entre las existentes actualmente. Por ejemplo: Base de datos JAFFE (Lyons, Akemastu, Kamachi, & Gyoba, 1998), CK (Kanade, Cohn, & Tian, 2000), CK+(Lucey, Cohn, Kanade, Saragih, & Ambadar,

2010), RAF-DB(Li, Deng, & Du, 2017), FER2013 (Challenges in representation learning, s.f.) u otros.

1.6 Organización de la tesis

El presente trabajo de tesis está compuesto por cinco capítulos, que serán descritos brevemente a continuación:

En el capítulo II, se presenta el marco teórico donde se describen los conceptos y términos relacionados al tema de tesis y serán mencionados a lo largo de esta.

Luego, en el capítulo III se presenta el estado del arte metodológico donde se dan a conocer los trabajos realizados previos a la presente tesis donde se proponen diversos métodos, técnicas y sistemas para dar solución al problema descrito en este trabajo.

El capítulo IV presenta el aporte teórico.

El capítulo V presenta la implementación donde se detalla la construcción de la solución.

Finalmente, el capítulo VI presenta la discusión, conclusiones y recomendaciones.

Capítulo II Marco teórico

En el presente capítulo se elaboró un glosario de términos que son usados frecuentemente en el reconocimiento de emociones y el reconocimiento automático de expresiones faciales. Las definiciones se han obtenido de fuentes confiables tales como: Artículos científicos y páginas web confiables, además para cada término se ha seleccionado los conceptos más adecuados al contexto del presente trabajo.

2.1 Emoción

La Real Academia Española define que la emoción como: “Alteración del ánimo intensa y pasajera, agradable o penosa, que va acompañada de cierta conmoción somática”(Española, s.f.).

2.2 Expresión facial

Acciones de todo tipo, si acompañan regularmente al estado de la mente, son reconocidos como expresivos (Darwin, 1872). Los movimientos expresivos de la cara y el cuerpo han sido de mucha importancia junto al lenguaje para la comunicación entre miembros de una misma tribu y por lo tanto al desarrollo del hombre (Darwin, 1872). Se puede observar algunas expresiones faciales en la Figura 1.



Figura 1: Ejemplo de expresiones faciales en las imágenes de la base de datos CK+

Fuente: (Lucey, Cohn, Kanade, Saragih, & Ambadar, 2010).

2.3 Emociones básicas

El psicólogo Paul Ekman postuló que existen exhibiciones faciales universales y distinguibles para cada uno de los principales afectos. Estos son: Ira, asco, miedo, alegría, tristeza y sorpresa (Ekman, Sorenson, & Friesen, Pan-cultural elements in facial displays of emotion, 1969).

2.4 Reconocimiento de emociones faciales

El reconocimiento facial de la expresión emocional es la capacidad de todos los individuos de reconocer formas básicas de expresión afectiva, la cual aparece en los rostros de las personas (Iglesias, Del Castillo, & Muñoz, 2016).

El estudio de la expresión es difícil debido a que los movimientos son a menudo extremadamente leves y de naturaleza flotante (Darwin, 1872).

El reconocimiento exacto de las emociones de otras personas es importante para la interacción social exitosa (Connolly, Lefevre, Young, & Lewis, 2019).

2.5 Fases del reconocimiento de emociones automático

Los sistemas de detección de emociones humanas consisten en múltiples pasos como recuperación de imágenes, pre procesamiento de imágenes, segmentación de imágenes,

extracción de características, clasificación de la expresión facial, entrenamiento seguido de detección de emociones(Khan, Samyan, Ghani, Shahid, & Wahla, 2019).

2.6Preprocesamiento de imágenes

El preprocesamiento de imágenes puede incluir normalización geométrica, normalización de brillo, ecualización de histograma y filtrado de imágenes (Wang, Li, Song, & Rong, 2019).

Las imágenes en los sistemas de reconocimiento de emociones a menudo se preprocesan mediante el uso de normalización geométrica, ecualización de histograma y técnicas de normalización del grado de gris para reducir el ruido en la imagen y garantizan la uniformidad del tamaño, posición y calidad de la imagen (Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019).

2.7Extracción de características

La extracción de características es la parte más importante del reconocimiento de la expresión facial. Una efectiva extracción de características faciales pueden mejorar el desempeño en el reconocimiento; si las características extraídas de las expresiones hacen que la distancia dentro de la clase sea lo más pequeña posible y la distancia entre clases sea lo más grande posible, las funciones del clasificador serán menos demandadas(Wang, Li, Song, & Rong, 2019).

2.8Clasificación de expresiones faciales

Es una importante parte del reconocimiento facial, el diseño del clasificador de la expresión facial afecta en gran medida la precisión del reconocimiento de la expresión facial. La selección y aplicación del clasificador es importante en la determinación del resultado final. El clasificador debe tener una alta eficiencia computacional y poderosa

capacidad para manejar un gran número de datos (Wang, Li, Song, & Rong, 2019). En la figura 2 se puede observar la clasificación de expresiones faciales.



Figura 2: Ejemplo de de clasificación de expresiones faciales

Fuente: (Khan, Samyan, Ghani, Shahid, & Wahla, 2019).

2.9 Base de datos de expresiones faciales

Estos conjuntos de datos se componen de videos e imágenes. Algunas contienen imágenes que se descargan de la web y otras son generadas localmente. Actualmente pocas bases de datos están divididas en positivas y negativas, algunas bases de datos están modeladas en 3D usando un motor de modelado 3D y contienen una variedad de expresiones faciales así como diversidad en término de ángulo de visión y poses. La mayoría de bases de datos son públicas y proporcionan imágenes consistentes en términos de resolución y claridad (Khan, Samyan, Ghani, Shahid, & Wahla, 2019).

2.10 Red Neuronal Convolutacional (CNN)

La Red Neuronal Convolutacional es un método de aprendizaje supervisado que puede realizar la extracción de características y el proceso de clasificación simultáneamente; además puede descubrir automáticamente los múltiples niveles de representaciones en los datos.

Este método es muy utilizado para la clasificación de imágenes ya que puede mapear información más profunda. La Red Neuronal convolutacional incluye principalmente la

capa de convolución, la capa de agrupación y la capa de conexión completa. La capa de convolución es responsable de la extracción de características, la capa de agrupación se utiliza para la selección de características y la capa de conexión completa se usa para la clasificación(Wang, Li, Song, & Rong, 2019).

Las redes neuronales convolucionales son un tipo de red neuronal artificial de alimentación inspirada en el tejido de la corteza visual animal. Es una red neuronal multicapa supervisada capaz de extraer características prominentes y distintivas de las imágenes faciales, dicha característica tiene habilidades deseables de generalización para imágenes desconocidas(Wei, Jia, Feng, Chen, & Chu, 2019). En la Figura 3 se puede observar la arquitectura de una red neuronal convolucional.

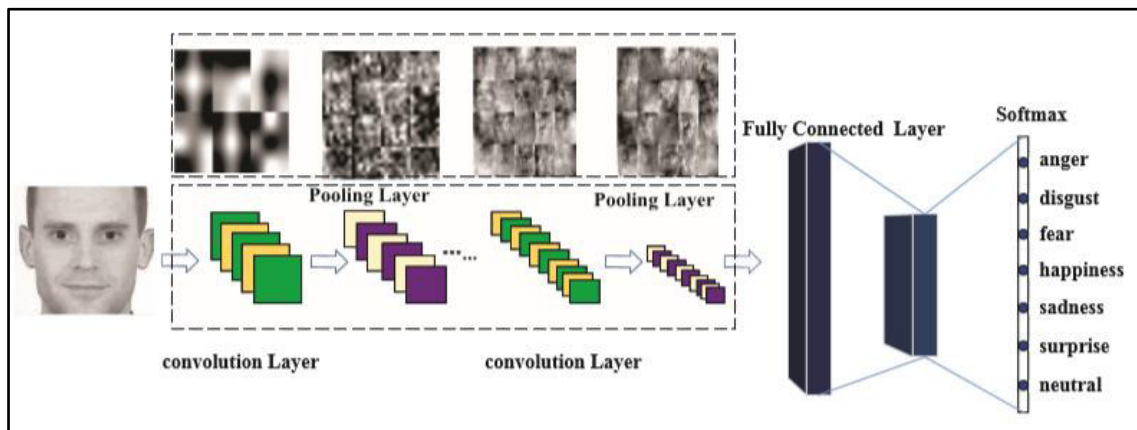


Figura 3: Estructura general de una implementación de Red Neuronal Convolucional

Fuente: (Wang, Li, Song, & Rong, 2019).

2.11 Base de datos de expresión facial femenina japonesa (JAFPE)

Base de datos de expresiones faciales publicada en 1998. La base de datos fue creada con la participación de 10 mujeres japonesas a las que se les pidió que realizaran una variedad de expresiones faciales (Wang, Li, Song, & Rong, 2019). Las 10 personas posaron tres o cuatro ejemplos de cada una de las seis expresiones faciales básicas(Felicidad, tristeza, sorpresa, enojo, disgusto y miedo) y neutral conformaron un total de 219 imágenes de expresiones faciales. Solo se empleó mujeres japonesas para practicidad del diseño experimental. Se puede observar algunas imágenes de la base de datos de expresiones faciales JAFPE en la Figura 4.

Cada mujer se tomó fotos mientras se miraban a través de una lámina de plástico semireflectable hacia la cámara. El cabello se recogió del rostro para exponer todas las zonas expresivas del rostro. Las luces del tungsteno fueron colocadas para crear una iluminación en la cara. Una caja encerraba la región entre la cámara y la lámina de plástico para reducir la retro-reflexión (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

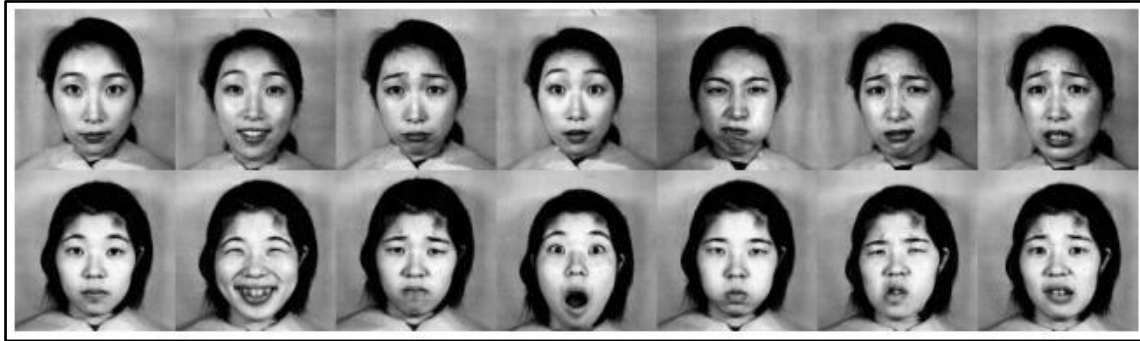


Figura 4: Imágenes de base de datos JAFFE

Fuente: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

En la Figura 5 se visualiza el aparato utilizado para fotografiar las expresiones faciales de la base de datos JAFFE.

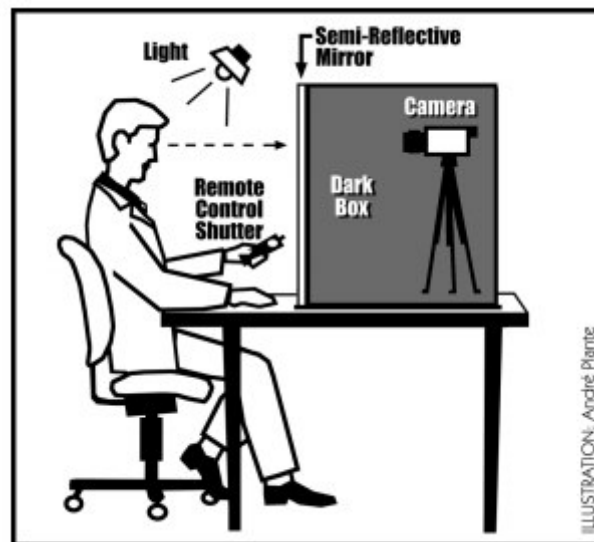


Figura 5: Aparato con el cual se fotografió las expresiones faciales en JAFFE

Fuente: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

2.12 Base de datos Cohn-Kanade (CK)

El comportamiento facial se registró en 210 adultos entre las edades de 18 y 50 años. El 69% eran mujeres, 31% hombres, 81% europeos, 13% afroamericanos y 6% otros grupos. Los adultos fueron observados en una sala equipada con una silla para sentarse y dos cámaras cada una conectada a una videograbadora. Una de las cámaras se localizó directamente frente al sujeto y la otra estaba colocada 30 grados a la derecha del sujeto. Para aproximadamente un tercio de los sujetos, la iluminación del cuarto aumentó mediante una lámpara de alta intensidad. Para los otros dos tercios, se utilizaron dos lámparas de alta intensidad con paraguas reflectantes para proporcionar iluminación uniforme (Kanade, Cohn, & Tian, 2000).

Las imágenes de expresión faciales se tomaron bajo un entorno de iluminación complejo (Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019).

CK incluyó 486 secuencias de códigos FACS de 97 sujetos (Lucey, Cohn, Kanade, Saragih, & Ambadar, 2010).

2.13 Base de datos Extended Cohn-Kanade (CK+)

Es una extensión de la base de datos Cohn-Kanade, que se lanzó en 2010. Para la distribución CK+, se ha aumentado aún más el conjunto de datos comparado con CK, para incluir 593 secuencias de 123 sujetos (Un adicional de 107 secuencias (22%) y 26 sujetos (27%)). La secuencia de imágenes varía en duración; es decir, de 10 a 60 fotogramas e incorporar el comienzo (El cual es también el marco neutral.) para alcanzar la formación máxima de las expresiones faciales. El comportamiento facial de 210 adultos se registró usando dos cámaras sincronizadas por Hardware. Los participantes tenían entre 18 y 50 años 69% mujeres, 81% euro-americanos, 13% afroamericanos y 6% de otros grupos (Lucey, Cohn, Kanade, Saragih, & Ambadar, 2010). En la Figura 6 se muestra ejemplos de imágenes de la base de datos CK+.



Figura 6: Imágenes de la base de datos CK+

Fuente: (Lucey, Cohn, Kanade, Saragih, & Ambadar, 2010)

2.14 Base de datos FER-2013

La base de datos incluye 35,887 diferentes imágenes. El conjunto de entrenamiento consta de 28,709 ejemplos. Hay dos subconjuntos de prueba: Pública y privada (Wang, Li, Song, & Rong, 2019), (Challenges in representation learning, s.f.).

2.15 Base de datos Real-world Affective Faces (RAF-DB)

Base de datos de rostros afectivos del mundo real, es una base de datos de expresión facial a gran escala. Consta de aproximadamente 30,000 imágenes faciales. Tiene dos carpetas: Básica y compuesta (Wang, Li, Song, & Rong, 2019).

Al principio las URL de las imágenes recolectadas de Flickr se ingresaron en un descargador automático de imágenes de código abierto en lotes. Teniendo en cuenta que los resultados devueltos por el API de búsqueda de imágenes de Flickr estaban bien estructurados en formato XML por lo cual las URLs se pudieron analizar fácilmente luego se usó un conjunto de palabras clave (Por ejemplo: Sonrisa, llanto, risa, miedo, rabia, asustado, aterrorizado, conmovido, asombrado, disgustado, inexpresivo.) para seleccionar imágenes relacionadas con las seis emociones básicas más la neutra. La Figura 7 muestra el proceso de construcción de la base de datos y en la Figura 8 se observa un ejemplo de las seis clases de emociones clásicas y las 20 clases de

emociones que componen la base de datos RAF-DB. El detalle de la distribución de imágenes es descrita a cada clase de expresión(Li, Deng, & Du, 2017). .

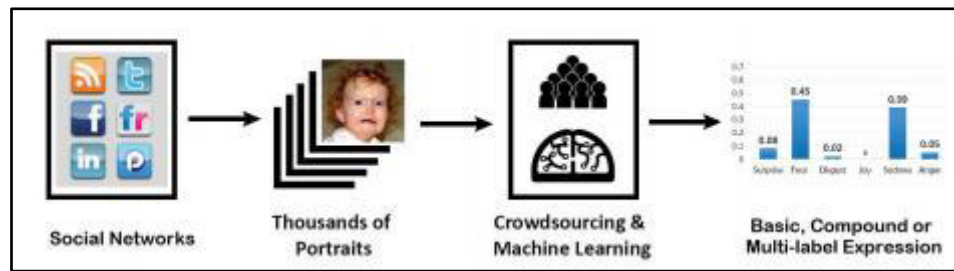


Figura 7: Proceso de construcción de la base de datos RAF-DB

Fuente: (Li, Deng, & Du, 2017)



Figura 8: Ejemplo de imágenes de las seis emociones básicas y las 20 clases de emociones de la base de datos RAF-DB

Fuente: (Li, Deng, & Du, 2017)

2.16 Ecualización de histograma

El histograma de imagen es un gráfico que representa la escala de grises de cada nivel en la imagen y su correspondencia de frecuencia. La abscisa indica el nivel de gris de los diferentes niveles de la imagen y la ordenada denota la frecuencia correspondiente de la escala de grises.

El valor del píxel varía de 0 a 255. La imagen en escala de grises es una imagen en la que el valor de cada píxel solo contiene información de intensidad, varía desde el negro de baja intensidad al blanco con mayor intensidad.

Su fórmula del cálculo de histograma es la siguiente:

$$P(k) = \frac{n_k}{N} \quad (1)$$

Fuente: (Li, Wang, Deng, & Xin, 2019).

Donde:

N: Se refiere al número total de píxeles en la imagen.

n_k : Representa la cantidad de píxeles con una escala de grises k.

$P(k)$: Representa la frecuencia con la que aparece el píxel.

La ecualización de histograma es un método capaz de ajustar el contraste de imágenes automáticamente.

La idea básica es determinar la función de transformación de gradación a través de la densidad de probabilidad de la escala de grises.

Con la transformación del histograma de grises de la imagen original a una forma uniformemente distribuida, la imagen original se estira de forma no lineal y además el valor concentrado de grises es redistribuido en toda la escala de grises.

Estas dos operaciones se llevan a cabo para mejorar el contraste en la imagen original.

A continuación la fórmula de transformación de la función de transformación y la función de densidad de probabilidad de la imagen original (Li, Wang, Deng, & Xin, 2019).

$$T(r) = \int_0^r p_r(r) dr \quad (2)$$

$$(0 \leq r \leq 1) \quad (3)$$

Fuente: (Li, Wang, Deng, & Xin, 2019).

Donde:

$T(r)$: Se refiere a la función de transformación

$P(r)$: Representa la función de densidad de probabilidad

A continuación, un ejemplo de la aplicación de la ecualización de histograma en una imagen. Se puede observar en las Figuras 9 y 11 el cambio en una imagen al aplicar ecualización de histograma y se puede observar en las Figuras 10 y 12 el cambio en el histograma de la imagen al aplicar la técnica previamente mencionada.

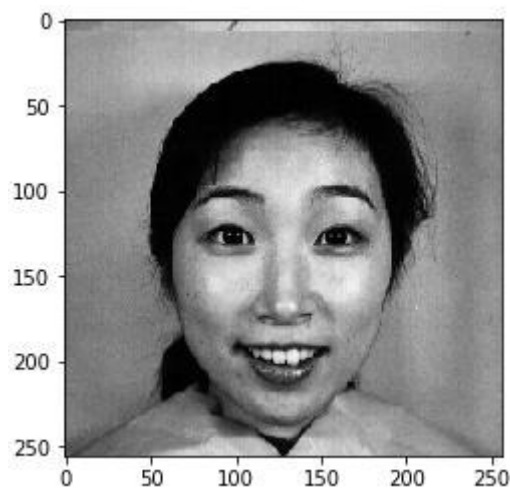


Figura 9: Imagen tomada de la base de datos JAFFE leída usando bibliotecaOpenCV

Fuente: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

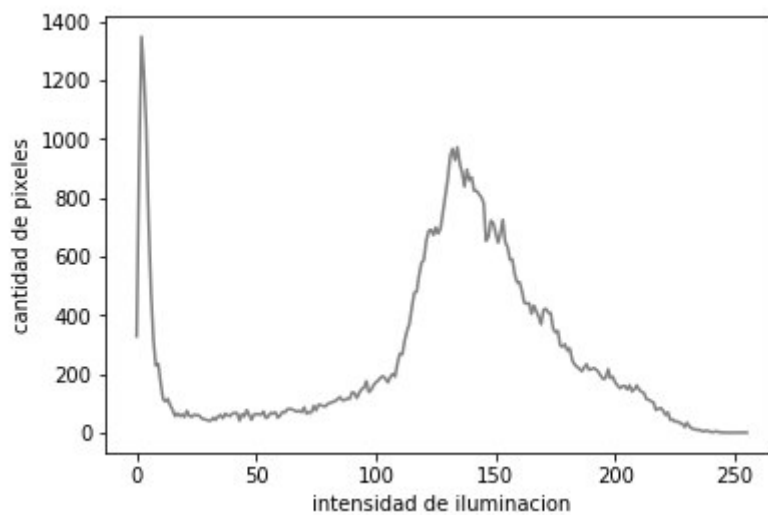


Figura 10: Histograma de la imagen original tomada de JAFFE elaborado usando Python

Fuente: Elaboración propia.

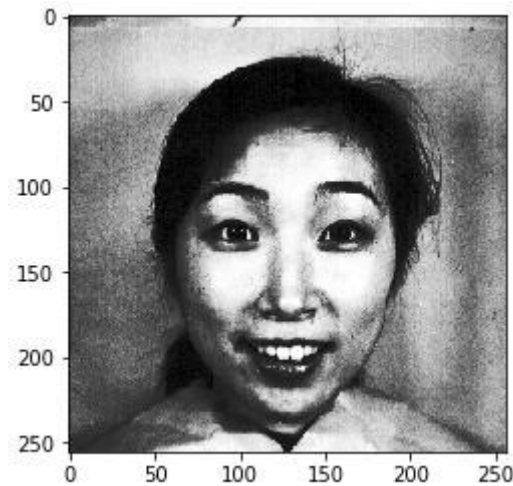


Figura 11: Imagen después de aplicar ecualización de histograma a una imagen tomada de la base de datos JAFFE, elaborado usando Python.

Fuente: (Lyons, Akemastu, Kamachi, & Gyoba, 1998)

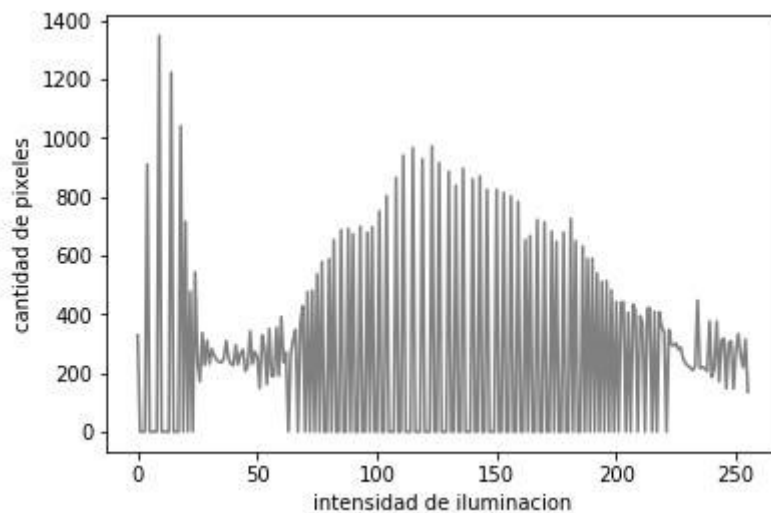


Figura 12: Histograma de la nueva imagen obtenida después de aplicar la técnica ecualización de histograma, elaborado usando Python

Fuente: Elaboración propia.

2.17 Transformada de wavelet discreta (DWT)

En DWT, los wavelets son discretizados usando escalar wavelet y traslaciones. Al realizar la transformación de wavelet discreta en cada imagen. De la imagen, se obtienen cuatro sub imágenes. Que incluyen muestras de paso bajo y muestras de paso alto. Las muestras de paso bajo (Sub banda LL) representan la aproximación de baja

resolución de la imagen original y las otras tres sub bandas están hechas de los coeficientes de la descomposición de wavelet y contienen los detalles insignificantes en la aproximación.

DWT tiene la ventaja de reducir la alta correlación entre los píxeles adyacentes por lo cual elimina la redundancia y provee una representación multiescala de la imagen (Lisungu & Sumbwanyambe, 2019). Se puede observar el proceso de descomposición que realiza el algoritmo 2D-DWT en la Figura 13.

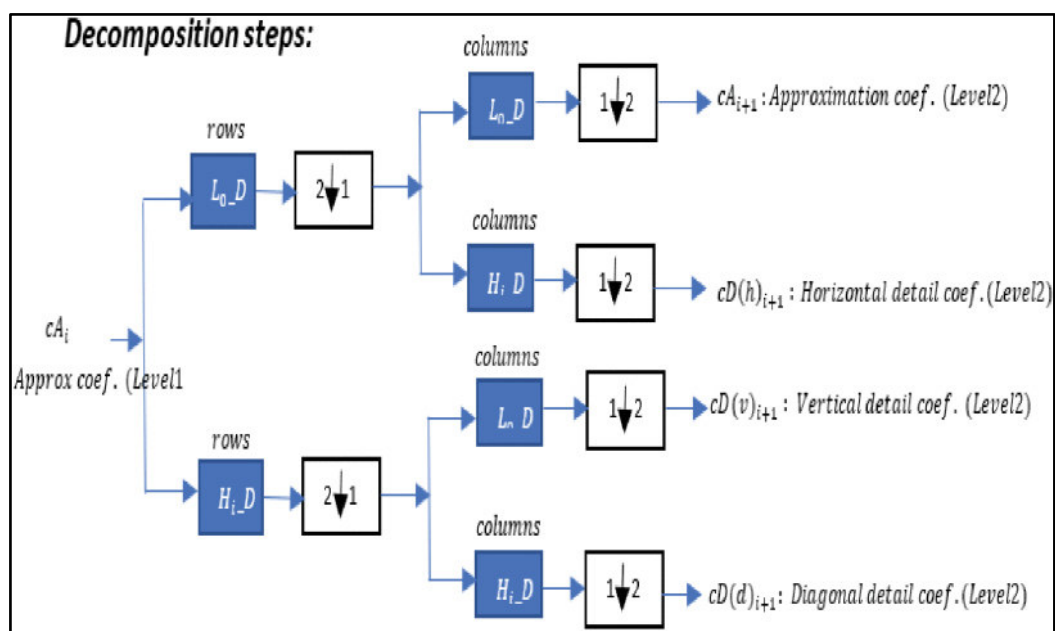


Figura 13: Proceso de descomposición de una imagen usando DWT

Fuente: (Lisungu & Sumbwanyambe, 2019).

2.18 Transformada discreta de wavelet bidimensional (2D-DWT)

El 2D DWT es una herramienta muy poderosa para extraer información, además suprime el ruido en la imagen.

El 2D-DWT descompone una imagen en cuatro sub bandas imágenes que son localizadas en frecuencia y orientación, denotadas por LL, LH, HL y HH.

La sub banda LL refleja la propiedad de suavizado de la imagen y este es usualmente llamado: imagen de baja frecuencia.

Las sub bandas LH, HL y HH reflejan las variaciones de intensidad de la imagen a lo largo de las direcciones horizontales, verticales y diagonales y suelen llamarse imágenes de alta frecuencia.

Los componentes de baja frecuencia contribuyen a la descripción global, mientras que los componentes de alta frecuencia contribuyen a los detalles. La data de la sub banda de baja frecuencia es muy comprimida respecto a la imagen general pero se mantiene la información principal (Li, y otros, 2019) (Mallat, 1989) (Navnidhi & Basha2, 2012).

La mayor parte de la energía de la imagen se concentra en el componente LL. Las sub bandas LL1, HL1, LH1 y HH1 son obtenidas después del primer nivel de descomposición. En el segundo nivel de descomposición, la sub imagen LL1 se descompone más en cuatro componentes: LL2, HL2, LH2 y HH2 (Lisungu & Sumbwanyambe, 2019). Se puede observar los coeficientes de la imagen al aplicar DWT y 2D-DWT en la Figura 14.

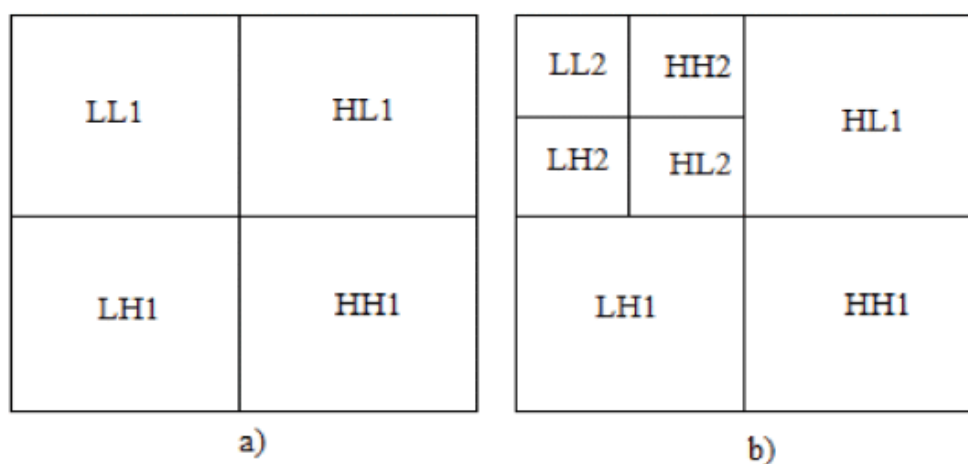


Figura 14: a) Nivel uno de descomposición wavelet b) Nivel dos de descomposición wavelet

Fuente: (Lisungu & Sumbwanyambe, 2019).

Se puede observar gráficamente los coeficientes obtenidos al aplicar 2D-DWT a una imagen de la base de datos JAFFE en la Figura 15.

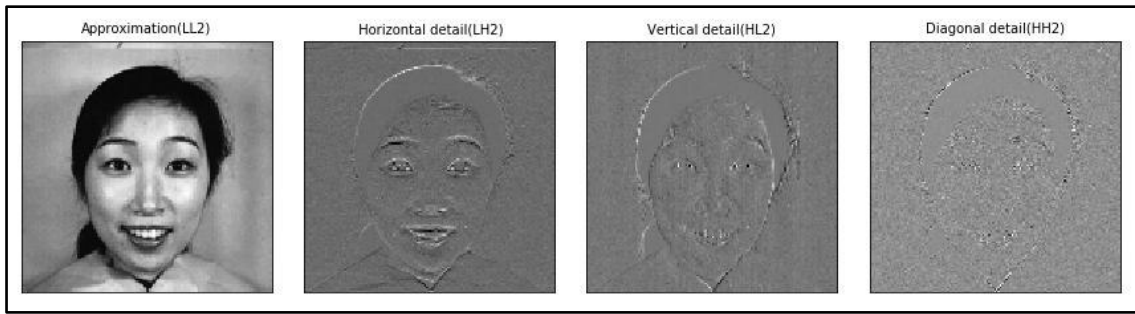


Figura 15: Coeficientes LL2, LH2, HL2 y HH2 de una imagen después de aplicar 2D-DWT a la imagen original, figura elaborada usando Python

Fuente de imagen original: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

2.19 Análisis de componentes principales (PCA)

El análisis de componentes principales es un método matemático para la reducción y extracción de características. Este método funciona eliminando los conjuntos de variables correlacionadas.

En otras palabras, este método se utiliza para reducir el número de variables de tal forma que pasemos de tener un número de variables a tener el mínimo número de variables nuevas. Las cuales representan a todas las variables originales de la manera más representativa posible. Se puede decir que PCA es una forma de identificar qué características son relevantes del conjunto de características linealmente dependientes.

Los componentes principales de un vector pueden ser usados por la característica de un vector largo. Esto reduce el número de dimensiones en la data, encuentra patrones en la data de alta dimensionalidad y visualiza la información de alta dimensionalidad.

El mejor espacio de baja dimensionalidad puede ser determinado por los mejores vectores propios de la matriz de covarianza de la data. Es decir, los vectores propios correspondientes a los valores propios más largos son también denominados: “Componentes principales” (Kumar, Kumar, Tripathi, & Kumar, 2019).

PCA reduce la dimensionalidad de la data usando transformación ortogonal. La matriz de datos es convertida a una matriz de puntajes, a una matriz de carga y a una matriz residual. La transformación convierte un conjunto de variables correlacionadas a un conjunto de variables no correlacionadas (Componentes principales). Como un

procedimiento estadístico multivariado no supervisado, PCA es ampliamente utilizado como una herramienta de exploración de datos. Particularmente, trazando los componentes principales(Gráfico de puntuaciones), los cluster pueden aparecer en el gráfico, los cuales son un indicativo de muestras con composición similar(Allegreta, y otros, 2020).

El algoritmo PCA es el siguiente:

Suponga $x_1, x_2, x_3, \dots, x_m$ son vectores $N \times 1$

Paso 1:

$$\bar{x} = \frac{1}{M} \sum_{i=1}^M x_i \quad (4)$$

Paso 2: Restar la media.

$$\phi_i = x_i - \hat{x}_i \text{ (Por ejemplo, centrar a cero)}$$

Paso 3: Formar la matriz $A = [\phi_1 \phi_2 \dots \phi_M] (N \times M)$, luego calcular:

$$C = \frac{1}{M} \sum_{i=1}^M \phi_i \phi_i^T = \frac{1}{M} A A^T \quad (5)$$

La matriz de covarianza de la muestra, $N \times N$, caracteriza la dispersión de los datos

Paso 4: Calcular los valores propios de:

$$C: \lambda_1 > \lambda_2 > \dots \lambda_N \quad (6)$$

Paso 5: Calcular los vectores propios de:

$$C: \mu_1 > \mu_2 > \dots \mu_N \quad (7)$$

Como C es simétrico, $\mu_1, \mu_2, \dots, \mu_N$ forman una base ortogonal (Es decir, cualquier vector x o en realidad $(x - \bar{x})$ puede escribirse como una combinación lineal de los vectores propios)

$$x - \bar{x} = b_1 \mu_1 + b_2 \mu_2 + \dots + b_N \mu_N = \sum_{i=1}^N b_i \mu_i \quad (8)$$

$$\text{Donde: } b_i = \frac{(x - \bar{x}) \cdot \mu_i}{\mu_i \mu_i} \quad (9)$$

Paso 6: (Paso de reducción de características) Preservar sólo los términos correspondiente a los valores propios K más altos:

$$\bar{x} - x = \sum_{i=1}^k b_i \mu_i \quad (10)$$

$$\text{Donde: } K \ll N \quad (11)$$

La transformación lineal $RN \rightarrow RK$ que realiza la reducción de la dimensionalidad o reducción de características es:

$$\begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_k \end{bmatrix} = \begin{bmatrix} \mu_1^T \\ \mu_2^T \\ \dots \\ \mu_K^T \end{bmatrix} (x - \bar{x}) = U^T (x - \bar{x}) \quad (12)$$

Fuente de fórmulas (4), (5), (6), (7), (8), (9), (10), (11) y (12): (Kumar, Kumar, Tripathi, & Kumar, 2019).

Como ejemplo, analizamos el conjunto de datos de Iris. Este dataset contiene 150 instancias de plantas iris cada una con cuatro características o atributos: Longitud del sépalo, ancho del sépalo, longitud del pétalo y ancho del pétalo (Fisher, D, & S, 1936).

Las plantas están categorizadas en tres clases (Setosa, Versicolor y Virginica) con 50 instancias de cada uno, donde cada clase refiere a un tipo de planta Iris (Guru, Suraj, & Manjunath, 2010). Los datos del conjunto de datos se pueden representar por una matriz de 150 filas por 4 columnas.

Al aplicar PCA se proyecta el conjunto de datos de Iris, inicialmente 4D, a un espacio de menor dimensionalidad (Dos dimensiones) como se muestra en la Figura 16.

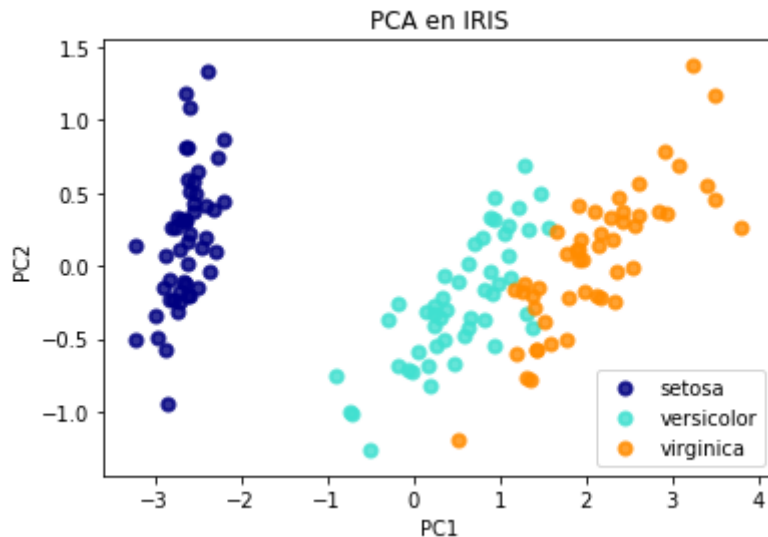


Figura 16: Conjunto de datos Iris proyectado usando PCA

Fuente:(Buitinck, y otros, 2013)

2.20 Máquinas de vectores de soporte o Máquinas de soporte vectorial

El algoritmo Máquinas de vectores de soporte(SVM) fue propuesto por el profesor Vapnik del instituto de investigación NEC en 1995 (Ming-wei, Yu-zhou, Jun-ping, Chen-bin, & Zong-hai, 2014).

Los sistemas de vectores de soporte son un conjunto de métodos de aprendizaje supervisado, los cuales son utilizados para clasificación y análisis de regresión.

Una máquina de vectores de soporte construye un hiperplano o un conjunto de hiperplanos para clasificar todas los datos, en un espacio de alta dimensionalidad. . Los vectores más cercanos al margen de clasificación son conocidos como vectores de soportes.

En un espacio bidimensional, el hiperplano es una línea

Las máquinas de vectores de soporte son muy populares y muchos los consideran el clasificador más práctico de implementar. Además, existen muchas herramientas que implementan las máquinas de vectores de soporte(Gove & Faytong, 2012).

Los datos del espacio original se asignan en el espacio de características de alta dimensión por un proceso de transformación no lineal, tal como se muestra en la Figura 17.

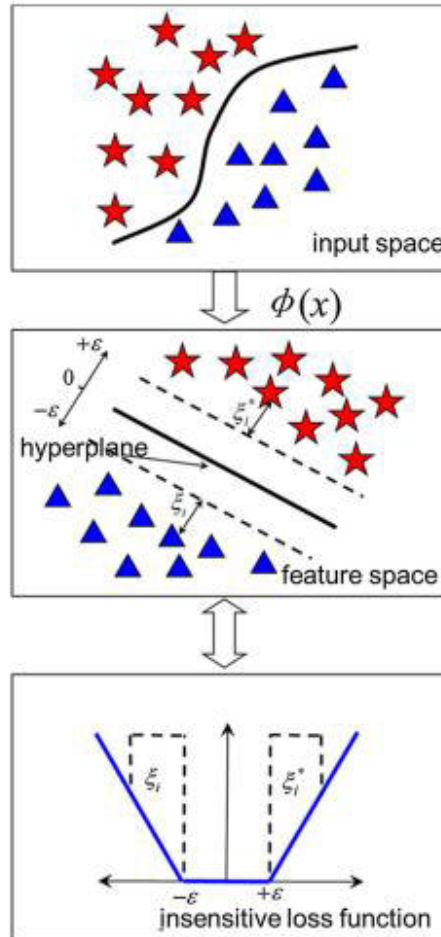


Figura 17: Proceso de transformación – Máquinas de Vectores de Soporte

Fuente: (Ma, Zhang, Xin, & Li, 2019).

A continuación, se explica el algoritmo SVM en el trabajo de (Navarro, 2018):

Se tienen vectores de entrada $x_i \in \mathbb{R}^d (i = 1, 2, \dots, n)$ correspondientes con las etiquetas $y_i \in \{-1, +1\}$ (Navarro, 2018).

Existe un plano separador cuya función es (Navarro, 2018):

$$\mathbf{w} \cdot \mathbf{x} + b = 0 \quad (13)$$

Donde $w \in \mathbb{R}^n$ es un vector normal que define la frontera, b es un umbral escalar y $\frac{|b|}{\|w\|}$ representa la distancia perpendicular desde el hiperplano separador hacia el origen (Navarro, 2018).

Mientras que la distancia del hiperplano al margen está dada por $d = \frac{1}{\|w\|}$. Dos hiperplanos paralelos pueden ser representados como (Navarro, 2018):

$$y_i \cdot (w \cdot x_i + b) \geq 1 \quad (14)$$

El objetivo del SVM es maximizar la distancia existente entre dos clases, por la amplitud del margen entre dos hiperplanos paralelos es $d = \frac{2}{\|w\|}$, por lo tanto para el caso de tratarse de datos linealmente separables se puede encontrar el hiperplano óptimo resolviendo el siguiente problema de optimización cuadrático (Navarro, 2018):

$$\min \frac{1}{2} \|w\|^2 \text{ sujeto a } y_i \cdot (w \cdot x_i + b) \geq 1 \quad (15)$$

Utilizando multiplicadores de lagangre α_i ($i = 1, 2, \dots, n$) para la restricción, el problema primal de la ecuación 16 se convierte a encontrar el punto de silla de Lagangre. Por lo tanto se convierte en un problema dual de la siguiente forma (Navarro, 2018):

$$\max L(\alpha) = \sum_i^l \alpha_i - \frac{1}{2} \sum_{i,j}^l \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (16)$$

$$\text{sujeto a: } \sum_{i=1}^n \alpha_i y_i = 0, \alpha_i \geq 0 \quad (17)$$

Resolviendo las condiciones de Karush-Kuhn-Tucker (KKT), se tiene (Navarro, 2018):

$$\alpha_i [y_i (w \cdot x_i + b) - 1] = 0 \quad (18)$$

Si $\alpha_i > 0$, los puntos correspondientes a los datos son llamados vectores de soporte. Por lo tanto, la solución óptima para el vector normal está dada por (Navarro, 2018):

$$w^* = \sum_{i=1}^N \alpha_i y_i x_i \quad (19)$$

Donde N es el número de vectores de soporte. De la ecuación 18, eligiendo cualquier vector de soporte (x_k, y_k) , se obtiene $b^* = y_k - w^* \cdot x_k$. Después (w^*, b^*) es determinado, a partir de la ecuación 19, el hiperplano separador óptimo puede denotarse como (Navarro, 2018):

$$f(x) = \text{sgn}(\sum_{i=1}^N \alpha_i y_i (x \cdot x_i) + b^*) \quad (20)$$

$$\text{Donde } \text{sgn}(\cdot) \text{ es la función signo, y } x \in \begin{cases} +1 & \text{si } f(x) > 0 \\ -1 & \text{si } f(x) < 0 \end{cases} \quad (21)$$

Fuente de fórmulas (13), (14), (15), (16), (17), (18), (19), (20) y (21): (Navarro, 2018)

Capítulo III Estado del arte metodológico

En el presente capítulo se presentarán los artículos con propuestas de solución al problema planteado en este trabajo de tesis.

3.1 Artículos

1. El artículo (Wang, Li, Song, & Rong, 2019) propuso un nuevo modelo para mejorar la precisión en el reconocimiento de expresiones faciales.

Este modelo utilizó una red neuronal convolucional (CNN) simple para extraer las características ya que este tipo de red puede mapear información más profunda con lo cual se mejora la precisión.

Se combinó las características extraídas por la red con un algoritmo de bosque aleatorio modificado; dicho algoritmo se colocó en la última capa de la red porque cuando los datos originales alcanzan completamente la capa de conexión, la naturaleza de los datos sin procesar se distorsionaba.

Se escogió utilizar el clasificador C4.5 por su alta eficiencia computacional y su capacidad de procesar gran cantidad de datos. Sin embargo, un solo clasificador es propenso a sobreajustar y su habilidad de generalización es débil por lo que se modificó para mejorar la tasa de ganancia de información.

Este trabajo utilizó cuatro bases de datos: JAFFE, CK+, FER-2013 (Se utilizó el conjunto de pruebas públicas para seleccionar el modelo CNN óptimo y el conjunto de pruebas privado para verificar la tasa de precisión del modelo CNN óptimo) y RAF-DB.

Finalmente se comprobó que el modelo propuesto tiene ventajas en el reconocimiento de emociones sobre la misma base de datos JAFFE en comparación con otros métodos que se basaron en un árbol de decisión o implementaron un bosque aleatorio sin modificaciones (Mollahosseini, Chan, & Mahoor, 2016), (Wen, Hou, Li, Jiang, & Xun, 2017), (Lucey, Cohn, Kanade, Saragih, & Ambadar, 2010), (Shan, Gong, & McOwan, 2009) y (Liu, Li, Shan, & Chen, 2015).

Sin embargo, este artículo no realizó experimentos tomando en cuenta condiciones especiales como la oclusión. Cabe mencionar que la red neuronal

convolucional requiere recolectar tantas muestras como sea posible para entrenarla y obtener buen rendimiento. Se muestra la arquitectura del método propuesto en la Figura 18.

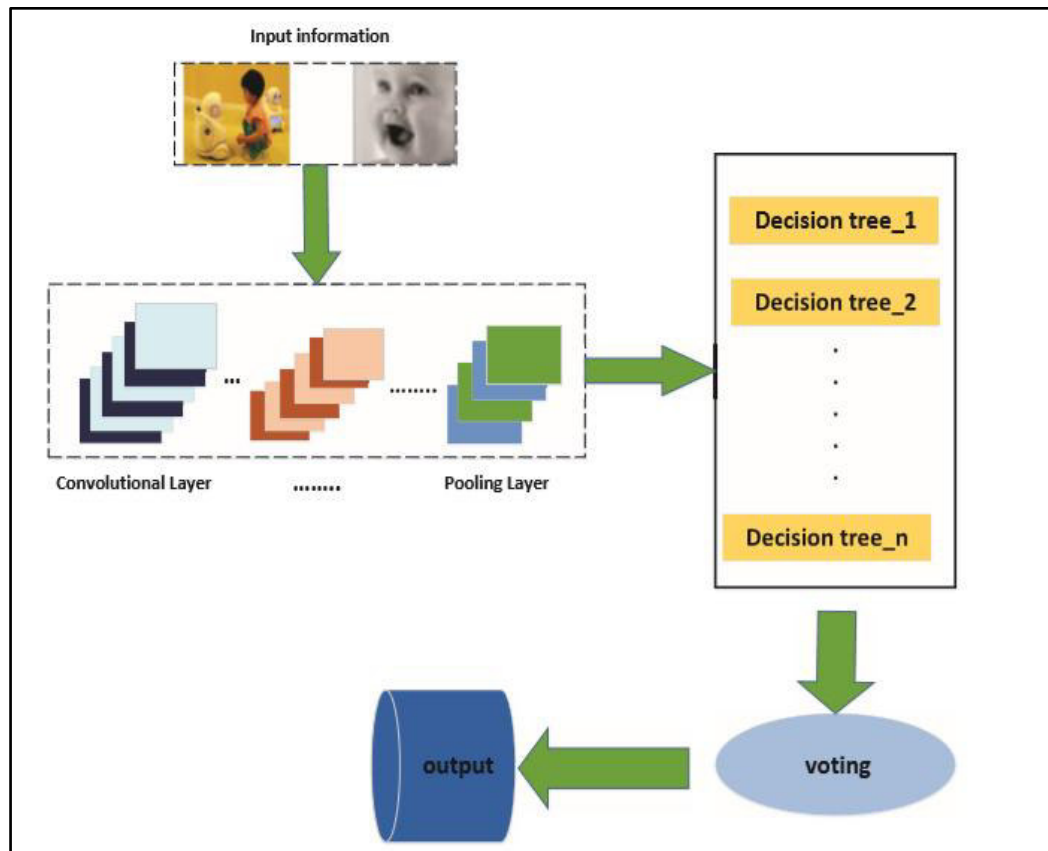


Figura 18: Estructura del modelo propuesto

Fuente: (Wang, Li, Song, & Rong, 2019).

2. El artículo (Wei, Jia, Feng, Chen, & Chu, 2019) propuso un novedoso método que emplea una estrategia multimodal para extraer características emocionales de las expresiones faciales.

La idea principal es combinar dos tipos diferentes de características: Características empíricas de bajo nivel (Selecciona puntos clave del rostro más discriminatorios para el reconocimiento de expresiones faciales. Se seleccionó 59 puntos de las cejas, ojo, nariz y boca) y características de autoaprendizaje de alto nivel extraídas mediante redes neuronales convolucionales (CNN).

Este método propuesto basado en redes neuronales convolucionales está configurado con ocho capas convolucionales y cuatro capas de agrupación;

además se le añadieron pequeños filtros en todas las capas con el fin de mejorar el poder diferenciador del modelo, reducir el número de parámetros y características. Dada la fusión de características se obtiene un único vector de características para cada imagen facial. Finalmente se aplicó SVM mediante el uso del paquete gratuito LIBSVM.

Los experimentos se realizaron con un conjunto de datos Cohn-Kanade y se aplicó una técnica para aumentar artificialmente el conjunto de datos con lo que se obtuvo finalmente un total de 1592 imágenes en el conjunto de imágenes.

La tasa de reconocimiento obtenida fue 93% con lo que mostró buen rendimiento en términos de tasa de reconocimiento de emociones. Y comparado con otros dos modelos (Happy & Routray, 2015), (Liu, Li, Shan, & Chen, 2015) y (Minh Vo, Sugimoto, & Hoang Le, 2016) que utilizaban otros métodos para extraer características, la presente propuesta demostró mejores resultados.

Sin embargo el conjunto de datos utilizado para las pruebas tenían muchas restricciones. Como trabajos futuros se plantea que el modelo propuesto utilice imágenes faciales de baja calidad, sin áreas faciales claves. Se muestra el diagrama de flujo del método propuesto en la Figura 19.

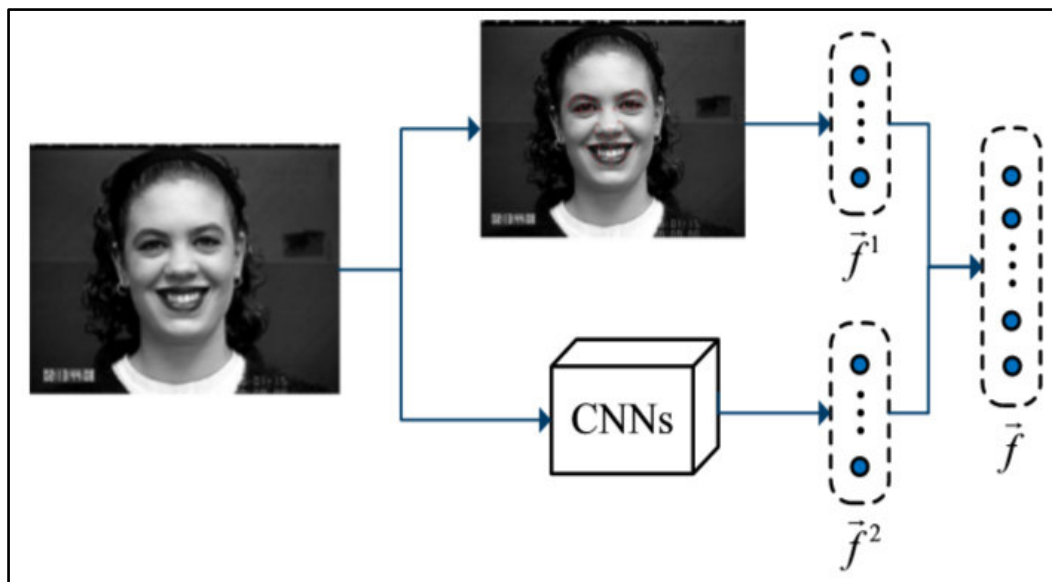


Figura 19: Diagrama de flujo del marco de representación de la expresión facial multimodal propuesto

Fuente: (Wei, Jia, Feng, Chen, & Chu, 2019).

3. El artículo (Chang & Ryoo, 2017) propuso un nuevo método para captar la emoción en una imagen facial.

Para la extracción de características el sistema se subdividió en fase de entrenamiento y fase de clasificación.

En la fase de entrenamiento, el área facial de la imagen se extrae utilizando puntos de referencia (Utilizó Active Shape Model o ASM); para reducir la dimensión de las características y clasificarlas se utilizó Fisherfaces. Después de esto, las características faciales finales son obtenidas mediante la imagen proyectada y los puntos de referencia de la imagen normalizada.

En la fase de clasificación, se utilizan nuevas imágenes faciales para clasificar las emociones. Al igual que en el entrenamiento se obtienen primero los puntos de referencia, luego las características faciales son producidos a mediante fisherface. Finalmente se extrae la emoción facial utilizando SVM.

Este nuevo sistema fue desarrollado en sistema operativo Windows con lenguaje C++, openCV y librerías stasm. También se implementó en sistema móvil (Android), donde se obtuvo el reconocimiento en tiempo real con solo presionar el botón de inicio.

Los experimentos se realizaron sobre tres bases de datos: Cohn-Kanade, JAFFE y una base de datos de expresiones faciales auto-fotografiadas. Para el entrenamiento se utilizaron 219,129 y 234 imágenes de las bases de datos mencionadas respectivamente y para la clasificación: 87, 57 y 94 imágenes respectivamente.

El nuevo sistema superó en casi un 12% a los resultados obtenidos por el algoritmo Eigenfaces (Lucey, Cohn, Kanade, Saragih, & Ambadar, 2010) y (Lyons, Akamatsu, Kamachi, & Gyoba, 1998). También mostró un resultado avanzado para las imágenes ásperas y auto-fotografiadas, con lo que se demuestra un alto valor agregado en el reconocimiento de emociones humanas en contenidos actuales.

Como trabajos futuros el artículo plantea que podría utilizarse un método avanzado de clasificación de características y un sistema de aprendizaje automático para mejorar la precisión del reconocimiento. Asimismo propone no solo usar las expresiones faciales para reconocer la emoción humana, sino también emplear el habla y el movimiento humano para el reconocimiento. Se observa la arquitectura del método propuesto en la Figura 20.

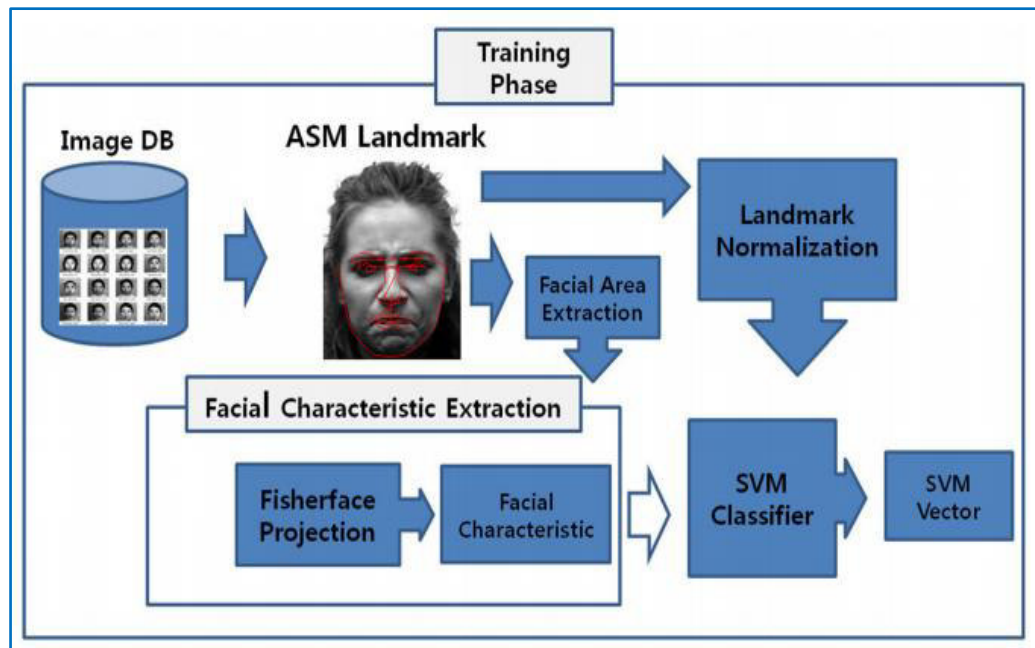


Figura 20: Procedimiento en la fase de entrenamiento

Fuente: (Chang & Ryoo, 2017).

4. El artículo (Happy, Dantcheva, & Bremond, 2019) propuso utilizar una red neuronal convolucional (CNN) con suavizado de etiquetas de manera supervisada y sintonizar los pesos CNN con datos etiquetados y sin etiquetar simultáneamente. Este modelo utilizó una fracción de datos no etiquetados junto con las muestras etiquetadas para actualizar la red; al realizar el suavizado sucesivo de etiquetas en los datos etiquetados, el modelo aprende las intensidades de expresión correctamente con lo cual se mejora el problema de sobre ajuste que se hace evidente en el caso de transición de expresiones.

Los experimentos se realizaron en cuatro conjuntos de datos: CK+, RaFD, lifespan y FER-2013.

Los pasos de preprocesamiento de imágenes implican la detección de rostro y alineación de rostro para posicionar ambos ojos a una distancia paralela al eje horizontal.

Esta propuesta demostró gran rendimiento comparado con otras propuestas que también usan CNN (Liu, Shan, Wang, & Chen, 2014), (Sikka, Sharma, & Bartlett, 2016), (Meng, Liu, Cai, Han, & Tong, 2017), (Liu, Han, Meng, & Tong, 2014), (Ding, Zhou, & Chellappa, 2017), (Jung, Lee, Yim, Park, & Kim, 2015) y (Zhao, y otros, 2016). Además el sistema logró aprender diferentes intensidades

de expresión y reconoció 7 clases de expresiones faciales. Se observa la arquitectura del método propuesto en la Figura 21.

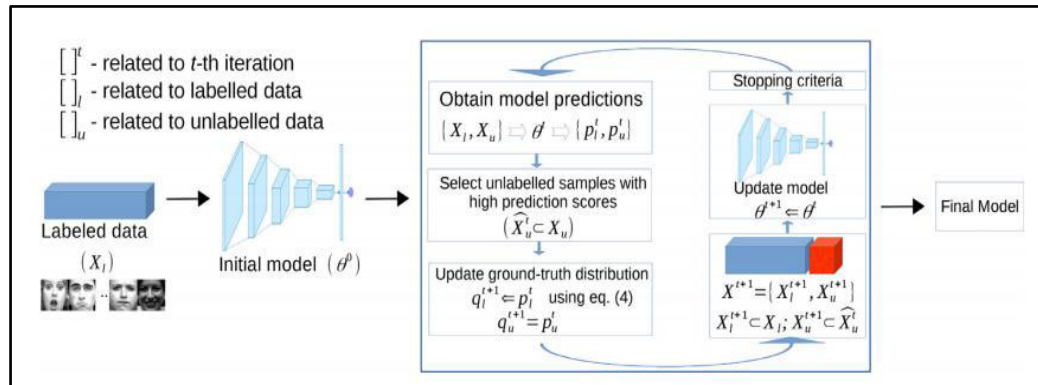


Figura 21: Flujo de trabajo del modelo propuesto

Fuente: (Happy, Dantcheva, & Bremond, 2019).

- El artículo (Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019) propuso un método para la identificación de expresiones faciales. Este método combina el algoritmo de detección de rostros Viola-Jones, algoritmo de ecualización de histograma adaptativo (CLAHE), la transformada de wavelet discreta (DWT) y red neuronal convolucional profunda.

El algoritmo Viola-Jones se utilizó para localizar la cara y sus partes, luego se mejora la imagen utilizando CLAHE.

La extracción de características se realiza mediante DWT. Finalmente las características extraídas se usan para entrenar la red CNN.

Los experimentos se realizaron en las bases de datos CK+ y JAFFE.

El método propuesto proporciona una alta precisión de reconocimiento de 99.2% para disgusto, 98.9% para sorpresa, felicidad y tristeza. Mientras que las emociones de ira, neutralidad y miedo también obtuvieron un alto nivel de precisión pero menos que las expresiones anteriormente mencionadas.

La base de datos JAFFE logró una precisión de reconocimiento del 98.63%.

En la base de datos CK+ se obtuvo una precisión del 97.05%. Los resultados de la precisión en CK+ fueron satisfactorios pero inferiores a los proporcionados con la base de datos JAFFE. Esto porque las imágenes en dicha base de datos

fueron capturadas en una condiciones de iluminación más difíciles. Sin embargo se demostró la efectividad del método.

En esta propuesta se elaboró un nuevo algoritmo denominado CLAHE el cual mejoró el contraste y brillo en las imágenes más que en otras técnicas (Cunha Santiago, Ren Tsang, & Cavalcanti, 2016), (Ai-Sumaidae , Dlay, Woo, & Chambers, 2015), (Hamester, Barros, & Wermter, 2015), (Nwosu, Wang, Unwala, Yang, & Zhang, 2017) y (Teixeira Lopes, De Aguiar, De Souza, & Oliveira-Santos, 2017). Se observa el proceso de extracción de características de Gabor en la Figura 22.

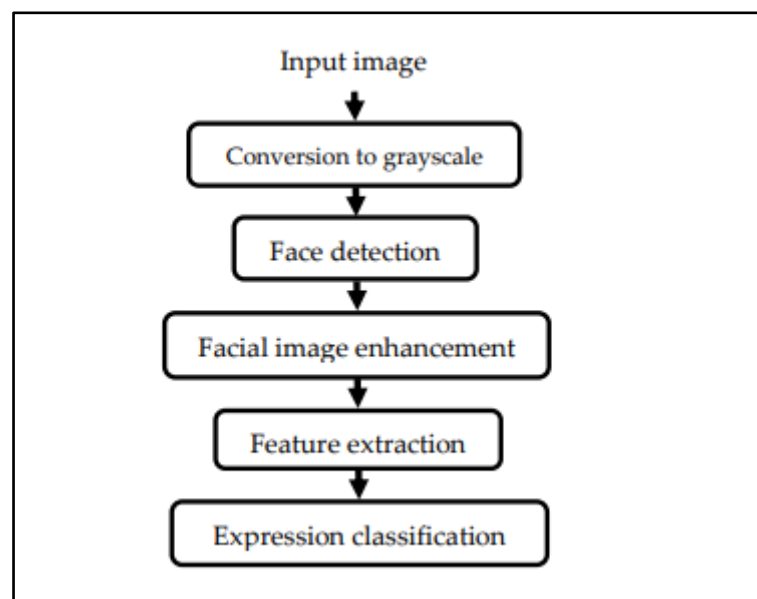


Figura 22: Proceso de extracción de características de Gabor

Fuente: (Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019).

IV Aporte teórico

4.1 Comparación de las tecnologías

En la Tabla 1, se presentan los artículos científicos considerados en el estado del arte.

ÍTEM	AUTORES	TÍTULO	REVISTA	AÑO DE PUBLICACIÓN
1	Yingying Wang, Yibin Li, Yong Song y Xuewen Rong.	Facial Expression Recognition Based on Random Forest and Convolutional Neural Network	Information	2019
2	Wei Wei, Qingxuan Jia, Yongli Feng, Gang Chen y Ming Chu.	Multi-modal facial expression feature based on deep-neural networks	Journal on Multimodal User Interfaces	2019
3	JaeKhun Chang y SeungTaek Ryoo.	Implementation of an improved facial emotion retrieval method in multimedia system	Multimed Tools Appl	2017
4	S L Happy, Antitza Dantcheva y Francois Bremond.	A Weakly Supervised learning technique for classifying facial expressions	Pattern Recognition Letter	2019
5	Ridha Ilyas Bendjillali, Mohammed Beladgham, Khaled Merit y Abdelmalik Taleb-Ahmed.	Improved Facial Expression Recognition Based on DWT Feature for Deep CNN	Electronics	2019

Tabla 1: Artículos analizados

Fuente: Elaboración propia.

En la Tabla 2, se comparan las ventajas y desventajas de los métodos descritos en el estado del arte.

ÍTEM	ARTÍCULO	VENTAJAS	DESVENTAJAS	MÉTODOS/TÉCNICAS UTILIZADAS
1	(Wang, Li, Song, & Rong, 2019)	• Extrae características más completas. • Soluciona los problemas de sobreajuste y generalización débil. • Demostró mayor tasa de reconocimient o comparado con técnicas que usan características artesanales. • Se disminuyó la alta configuración que se requiere en un modelo deep learning.	• No fue probado bajo condición de oclusión. • Para el entrenamien to de la red se necesita recolectar tantas muestras como sea posible.	Red neuronal CNN, clasificador C4.5, algoritmo de bosque aleatorio.
2	(Wei, Jia, Feng, Chen, & Chu, 2019)	• Aborda el problema de sobreajuste y desbalance de datos. • Obtuvo mejor tasa de reconocimient o comparado con métodos que solo utilizaban un tipo de características	• Se realizaron experimento s con imágenes que tenían muchas restricciones .	Red neuronal convolucional (CNN) y SVM.

		a través de las características multimodales. • Redujo el número de parámetros para la red neuronal convolucional sin afectar la efectividad de su aprendizaje. • Es robusto por su pequeña escala y sus pocos parámetros.		
3	(Chang & Ryoo, 2017)	• Demostró buenos resultados al trabajar con imágenes auto fotografiadas. • El sistema se pudo implementar en sistema Android y usarlo en tiempo real. • Comparado con otros el algoritmo que utiliza Eigenfaces, obtuvo casi 12% más de mejores resultados en casi todas las emociones.	• Solo se utilizó las expresiones faciales para reconocer la emoción.	Análisis de componentes principales(PCA), análisis discriminante lineal(LDA) y SVM.

4	(Happy, Dantcheva, & Bremond, 2019)	• Soluciona el problema de sobreajuste, el modelo aprende las intensidades de las emociones correctamente. • Demostró alto reconocimiento de emociones en los experimentos.	• Requiere complejas configuraciones.	Método basado en red neuronal convolucional con suavizado de etiquetas de manera supervisada.
5	(Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019)	• Utilizó el método CLAHE, el cual mejoró las condiciones de iluminación de las imágenes. • El método demostró ser efectivo, ya que obtuvo alta precisión en la base de datos JAFFE y CK+.	• Obtuvo mejores resultados con imágenes no complejas.	Método basado en algoritmo viola-jones, CLAHE, DWT y red neuronal convolucional.

Tabla 2: Comparación de ventajas, desventajas y métodos/técnicas utilizadas

Fuente: Elaboración propia.

En la Tabla 3, se muestran los criterios que se tomarán en cuenta para seleccionar un método.

Ítem	Criterios de evaluación	Definición
1	Complejidad	Es el grado de dificultad en la implementación del método.
2	Precisión	Es el grado de precisión con el que se reconoce la emoción.
3	Dominio de la técnica	Es el nivel de conocimiento que se posee sobre las técnicas y métodos requeridos para el desarrollo del sistema.

Tabla 3: Criterios de evaluación

Fuente: Elaboración propia.

En la Tabla 4, se observa los pesos asignados a cada criterio.

Criterios	Complejidad	Precisión	Dominio de la técnica
PESO	Alto:0 (0)	Alto: 20 (8)	Alto:20 (10)
	Medio:10 (4)	Medio: 10 (4)	Medio:10(5)
	Bajo:20 (8)	Bajo: 0 (0)	Bajo:0(0)

Tabla 4: Pesos de los criterios

Fuente: Elaboración propia.

En la Tabla 5, se muestran los puntajes alcanzados por los métodos del estado del arte.

Criterios de evaluación	Complejidad	Precisión	Dominio de la técnica	Puntaje
Método propuesto por (Wang, Li, Song, & Rong, 2019)	Alto(0)	Alto(8)	Bajo(0)	8
Método propuesto por (Wei, Jia, Feng, Chen, & Chu, 2019)	Alto(0)	Alto(8)	Bajo(0)	8
Método propuesto por (Chang & Ryoo, 2017)	Alto(0)	Alto(8)	Medio(5)	13
Método propuesto por (Happy, Dantcheva, & Bremond, 2019)	Alto(0)	Alto(8)	Bajo(0)	8
Método propuesto por (Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019)	Medio(4)	Alto(8)	Medio(5)	17

Tabla 5: Evaluación de métodos

Fuente: Elaboración propia.

4.2 Estrategia para el diseño de la solución y selección de tecnologías

Para el diseño de la solución se seguirá la siguiente estrategia:

- Definir las fases en que se dividirá la solución.
- Identificar el número y qué emociones se reconocerán.
- Elegir una base de datos de expresiones faciales que contengan imágenes con las etiquetas de emociones a reconocer.
- Seleccionar las técnicas, métodos y algoritmos más convenientes a aplicar en cada fase del sistema y lograr los objetivos planteados.

Diversos métodos fueron propuestos por los autores en el estado del arte, los cuales combinados apropiadamente podrían obtener óptimos resultados en el reconocimiento de emociones.

Para el diseño de la solución se emplea la estrategia observada en los artículos del estado el arte, la cual consiste en la comparación de técnicas existentes para escoger las que más se adecuen a los objetivos del trabajo de investigación.

Por lo tanto:

- Se mantendrán las fases de los sistemas de reconocimiento automático de emociones utilizados en la mayoría de propuestas: Pre procesamiento, extracción y reducción de características, entrenamiento y clasificación de la expresión facial.
- El sistema reconocerá siete emociones, las cuales serán: Felicidad, tristeza, disgusto, enojo, miedo, desagrado y neutral.
- Se utilizará la base de datos de expresiones faciales japonesa femenina (JAFFE) ya que es pública y contiene el número de imágenes necesarias para el entrenamiento y la clasificación.

- Después de realizar la comparación entre los métodos propuestos por los artículos del estado del arte, se vio conveniente y viable utilizar ecualización de histograma en el pre procesamiento para mejorar las condiciones de iluminación presente en las imágenes. Transformada discreta de wavelet bidimensional y análisis de componentes principales en la fase de extracción y reducción de características respectivamente. Finalmente se usará máquinas de vectores de soporte lineal en la fase de clasificación.

4.3 Metodología

Para la mejor comprensión del sistema propuesto, se presenta la arquitectura de la solución en la Figura 23.

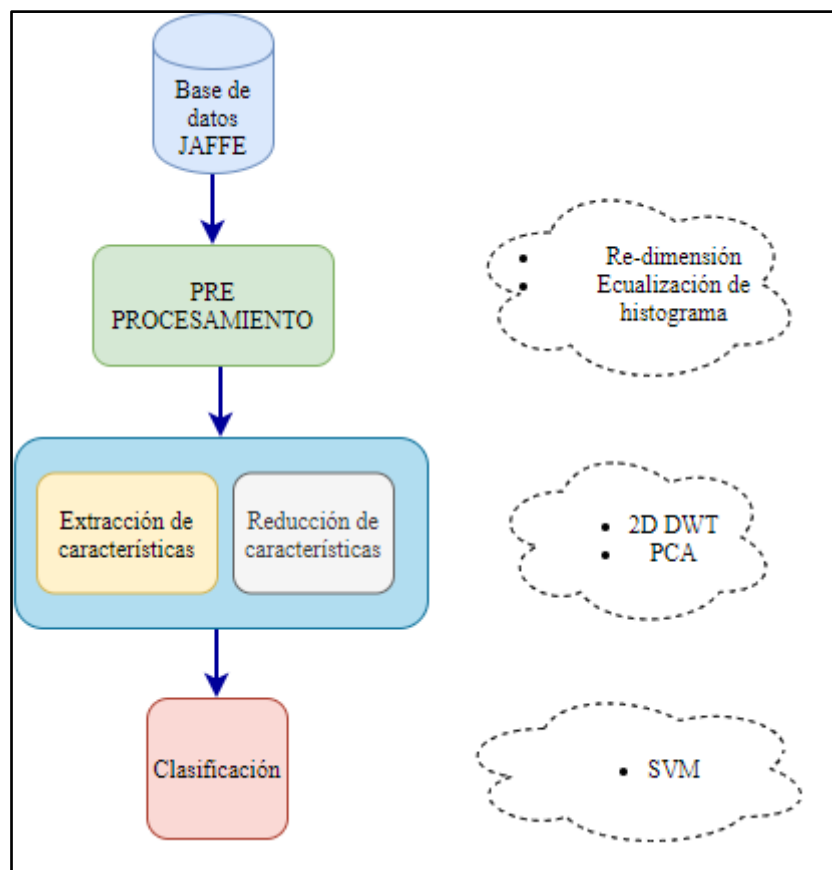


Figura 23: Arquitectura del sistema de reconocimiento de expresiones faciales propuesto

Fuente: Elaboración propia.

Ahora se explicará la metodología a seguir en cada fase del sistema propuesto de reconocimiento automático de emociones faciales:

4.3.1 Base de datos de expresiones faciales

La base de datos de expresiones faciales a usar será JAFFE debido a que se trata de una base de datos pública.

Se dividirá aleatoriamente el conjunto de datos: El 80% de las imágenes de la base de datos será utilizada para entrenar al sistema y el 20% restante se usarán para realizar las pruebas.

4.3.2 Preprocesamiento de imágenes

Como parte del preprocesamiento, primero se detectarán los ojos, luego se rotará la imagen de tal forma que la recta formada por los ojos sea paralela al eje x.

Se recortará la imagen para eliminar el fondo y reducir el ruido en la imagen. Con lo cual nos quedaremos con la región de interés que en este caso es la cara y se redimensiona el tamaño de la imagen a 96x128. Como paso final de la fase de preprocesamiento, se aplicará ecualización de histograma para reducir los efectos de diferencia de iluminación en cada imagen de la base de datos de expresiones faciales.

4.3.3 Extracción de características

En esta etapa aplicaremos dos algoritmos para la extracción de características.

Se aplicará algoritmo 2D-DWT a las imágenes de la base de datos de expresiones faciales: El coeficiente de aproximación de segundo nivel de descomposición (Mayor detalle en la sección 2.18 de la presente tesis) contendrá las características que representen la información esencial de las imágenes, de tal manera que podamos clasificarlas posteriormente.

Finalmente, se usará la técnica de análisis de componentes principales(PCA) para reducir la dimensionalidad de las características. De tal forma que en la siguiente etapa(Clasificación), el algoritmo pueda trabajar con esta información y obtener los mejores resultados posibles.

4.3.4 Clasificación

Para la presente etapa, se utilizarán Máquinas de Vectores de Soporte(SVM) para abordar la clasificación de las expresiones faciales en siete clases.

En este trabajo se aplica SVM con la configuración lineal para la separación de clases.

4.3.5 Entrenamiento del reconocedor

Consistirá en alimentar al modelo con las características extraídas del conjunto de imágenes de entrenamiento y sus respectivos valores conocidos (Etiquetas) con la finalidad de que pueda aprender.

4.3.6 Pruebas

Las pruebas se realizarán con un conjunto de datos conformado por el 20% del total de imágenes.

Las características extraídas de las imágenes de pruebas alimentarán al modelo previamente entrenado para poder predecir las emociones. Con el modelo ya determinado, se utilizará el algoritmo SVM para la predicción con lo cuál se podrá validar el modelo.

4.3.7 Validación del sistema

Para validar la efectividad de la solución propuesta, se tendrá en cuenta las siguientes métricas:

- **Exactitud:** Es la métrica de evaluación más usada en la práctica, ya sea para problemas de clasificación binaria o multi-clase. A través de la precisión, la calidad de la solución es evaluada basado en el porcentaje de predicciones correctas sobre el total de instancias (Hossin, Mohammad , N, & Sulaiman, 2015).

$$Exactitud = \frac{tp+tn}{tp+fp+tn+fn} \quad (22)$$

Donde:

tp: Verdadero positivo

fn: Falsosnegativo

tn: Verdadero negativo

fp: Falso positivo

Fuente: (Hossin, Mohammad , N, & Sulaiman, 2015)

- **Sensibilidad:** Indica la capacidad para diferenciar los casos positivos de los negativos. Esta métrica es usada para medir la fracción de casos positivos que fueron correctamente clasificados (Hossin, Mohammad , N, & Sulaiman, 2015).

$$sensibilidad = \frac{tp}{tp+fn} \quad (23)$$

Fuente: (Hossin, Mohammad , N, & Sulaiman, 2015)

- **Puntaje F1:** Esta medida representa la media armónica entre los valores de las métricas sensibilidad y precisión (Hossin, Mohammad , N, & Sulaiman, 2015).

$$Puntaje F1 = 2 \times \frac{precisi \acute{o}n \times sensibilidad}{precisi \acute{o}n + sensibilidad} \quad (24)$$

Fuente: (Hossin, Mohammad , N, & Sulaiman, 2015)

- **Precisión:** La precisión es usada para medir los casos positivos que fueron clasificados correctamente del total de predicciones de clase positiva (Hossin, Mohammad , N, & Sulaiman, 2015).

$$Precisi \acute{o}n = \frac{tp}{tp+fp} \quad (25)$$

Fuente: (Hossin, Mohammad , N, & Sulaiman, 2015)

- Soporte: Es el número de ocurrencias de clases.

4.4 Tecnologías a utilizar

4.4.1 Hardware

Computador: Asus Core i5, con memoria RAM de 8 Gb y sistema operativo Windows 10.

Para el desarrollo, pruebas e implementación de este proyecto se utilizará un computador con las siguientes características.

- Sistema operativo: Windows 10 Home 64-bit (10.0, Build 18362) (18362.19h1_release.190318-1202)
- Procesador: Intel(R) Core(TM) i5-6198DU CPU @ 2.30GHz (4 CPUs), ~2.4GHz
- Memoria: 8 GB RAM

4.4.2 Software

Se tomará como base la implementación Python(Github, 2019) basada en el artículo(Shih, Wang, & Chuang, 2008).

Para la implementación del sistema se utilizará:

Python, como lenguaje de programación (Rossum & G, 1995)ya que contiene librerías de inteligencia artificial con los métodos a implementar en la solución. Se observa su logo en la Figura 24.



Figura 24: Logo python

Se utilizará la aplicación web Jupyter Notebook para ejecutar el código Python. Se observa el logo en la Figura 25. Jupyter Notebook es open-source.



Figura 25: Logo Jupyter Notebook

Además, se importarán las siguientes librerías:

- OpenCV: Biblioteca de visión artificial (Bradski & G, 2000). Se observa el logo en la Figura 26.



Figura 26: Logo OpenCV

- Numpy: Biblioteca numérica (Van der Walt, Colbert, & Varoquaux, 2011).
Permite utilizar matrices N dimensionales. Se observa el logo en la Figura 27.



Figura 27: Logo Numpy

- Scikit learn: Librería para aprendizaje automático (Pedregosa, y otros, 2011).
Proporciona técnicas de aprendizaje automático supervisado y no supervisado.
Se observa el logo en la Figura 28.



Figura 28: Logo Scikit learn

- Matplotlib: Permite visualizar los datos(Hunter, 2007). Se observa el logo en la Figura 29.



Figura 29: Logo Matplotlib

- Dlib: Proporciona algoritmos de aprendizaje de máquina (King, 2009). Se observa el log en la Figura 30.



Figura 30: Logo Dlib

- Imutils: Contiene herramientas que permiten el procesamiento de imágenes.
- PyWavelets: Permite implementar wavelets(Lee, Gommers, Waselewski, Wohlfahrt, & O’Leary, 2019).
- Pandas: Proporciona herramientas para visualizar datos(McKinney, 2010). Se observa el logo en la Figura 31.



Figura 31: Logo Pandas

Capítulo V Implementación

En el presente capítulo se describe el desarrollo del sistema propuesto para el reconocimiento automático de emociones faciales.

5.1 Desarrollo de la solución

5.1.1 Preparación de la base de datos de expresiones faciales

Descargar la base de datos de expresiones faciales JAFFE de la siguiente página web:

<http://www.kasrl.org/jaffe.html>

Esta base de datos contiene 213 imágenes, por lo cual se utilizará 43 para las pruebas y 170 para el entrenamiento del sistema.

Luego, crear las subcarpetas “Reconocedor/jaffe” en el directorio Documents y copiar las imágenes de la base de datos descargada previamente. Se muestran las imágenes en el directorio local en la Figura 32.

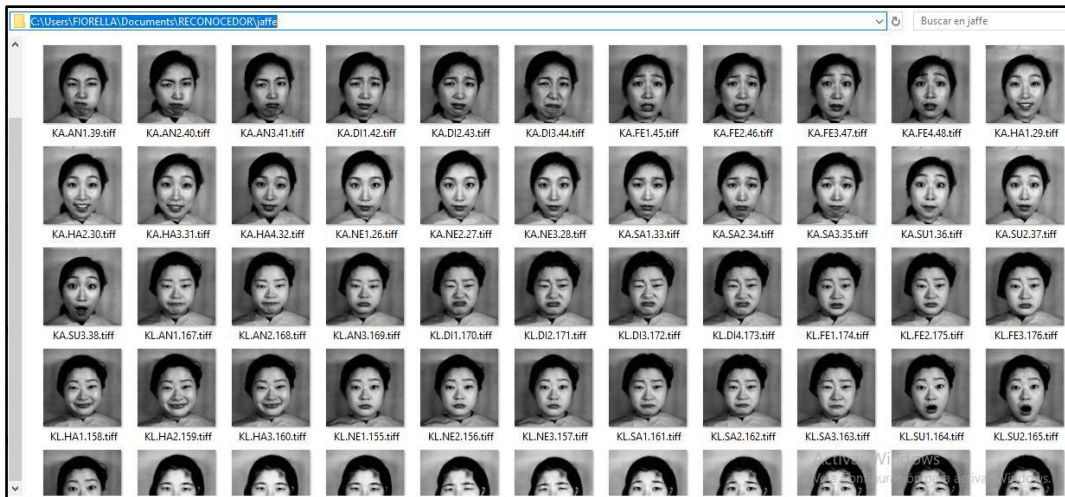


Figura 32: Imagen de la carpeta RECONOCEDOR donde se colocó las imágenes descargadas de la base de datos JAFFE

Fuente de las imágenes: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

5.1.2 Preparación del ambiente de desarrollo

En la ventana CMD, ejecutar el siguiente comando:

```
C:\Users\FIORELLA>pip install --upgrade pip
Collecting pip
  Downloading https://files.pythonhosted.org/packages/5c/e0/be401c003291b56e
fc55aeba6a80ab790d3d4cece2778288d65323009420/pip-19.1.1-py2.py3-none-any.whl
(1.4MB)
    100% |#####| 1.4MB 207kB/s
Installing collected packages: pip
  Found existing installation: pip 18.1
  Uninstalling pip-18.1:
    Successfully uninstalled pip-18.1
```

Instalar virtualenv, herramienta para crear ambientes virtuales.

```
C:\Users\FIORELLA>pip install virtualenv
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020.
Please upgrade your Python as Python 2.7 won't be maintained after that date.
A future version of pip will drop support for Python 2.7.
Collecting virtualenv
  Downloading https://files.pythonhosted.org/packages/ca/ee/8375c01412abe6ff462ec80970e6bb1c4308724d4366d7519627c98691ab/virtualenv-16.6.0-py2.py3-none-any.whl (2.0MB)
    |#####| 2.0MB 678kB/s
Installing collected packages: virtualenv
Successfully installed virtualenv-16.6.0
```

```
C:\Users\FIORELLA\Documents\RECONOCEDOR>virtualenv env
New python executable in C:\Users\FIORELLA\Documents\RECONOCEDOR\env\Scripts\python.exe
Installing setuptools, pip, wheel...
done.
```

Activar el ambiente virtual.

```
C:\Users\FIORELLA\Documents\RECONOCEDOR>env\Scripts\activate.bat

(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>
```

Instalar la librería PyWavelets.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install PyWavelets
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Collecting PyWavelets
  Downloading https://files.pythonhosted.org/packages/67/68/767910bcabbcd818b99d73c42f754cfa2f9cb4e1622275e61c0d7d8496/PyWavelets-1.0.3-cp27-none-win32.whl (4.1MB)
    |#####| 4.1MB 166kB/s
Collecting numpy>=1.9.1 (from PyWavelets)
  Downloading https://files.pythonhosted.org/packages/13/2d/0fa2e8de7022a4a39497f4a9e384b8b129dbc5d1b659f1043e21f6f0a48/numpy-1.16.4-cp27-cp27m-win32.whl (10.0MB)
    |#####| 10.0MB 142kB/s
Installing collected packages: numpy, PyWavelets
Successfully installed PyWavelets-1.0.3 numpy-1.16.4
```

Instalar la librería dlib.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install dlib-18.17.100-cp27-none-win32.whl
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Processing c:\users\fiorella\documents\reconocedor\dlib-18.17.100-cp27-none-win32.whl
Installing collected packages: dlib
Successfully installed dlib-18.17.100
```

Instalar la biblioteca OpenCV.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install opencv-python
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Collecting opencv-python
  Downloading https://files.pythonhosted.org/packages/89/80/b2a0e2350331336769e2b31449732eb415021c3694d6bed7413d85c66f/opencv_python-4.1.0.25-cp27-cp27m-win32.whl (25.7MB)
    |#####| 25.7MB 125kB/s
Requirement already satisfied: numpy>=1.11.1 in c:\users\fiorella\documents\reconocedor\env\lib\site-packages (from opencv-python) (1.16.4)
Installing collected packages: opencv-python
Successfully installed opencv-python-4.1.0.25
```

Instalar la librería imutils.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install imutils
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Collecting imutils
Installing collected packages: imutils
Successfully installed imutils-0.5.2
```

Instalar la librería Sklearn.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install sklearn
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Collecting sklearn
  Downloading https://files.pythonhosted.org/packages/1e/7a/dbb3be0ce9bd5c8b7e3d87328e79663f8b263b2b1bfa4774cb1147bfcd3f/sklearn-0.0.tar.gz
Collecting scikit-learn (from sklearn)
  Downloading https://files.pythonhosted.org/packages/b3/8f/f87cd37b2aaf15321af7344ad58d53181d8c3529b72838fe39d4290208b0/scikit_learn-0.20.3-cp27-cp27m-win32.whl (4.5MB)
[#####] 4.5MB 227kB/s
Collecting scipy>=0.13.3 (from scikit-learn->sklearn)
  Downloading https://files.pythonhosted.org/packages/df/dc/ec5ef4756b176ce40d1c8792aeb3130ac2d57b7d0a2e6a7e22f352c20911/scipy-1.2.1-cp27-cp27m-win32.whl (27.3MB)
[#####] 27.3MB 47kB/s
Requirement already satisfied: numpy>=1.8.2 in c:\users\fiorella\documents\reconocedor\env\lib\site-packages (from scikit-learn->sklearn) (1.16.4)
Building wheels for collected packages: sklearn
  Building wheel for sklearn (setup.py) ... done
  Stored in directory: C:\Users\FIORELLA\AppData\Local\pip\Cache\wheels\76\03\bb\589d421d27431bcd2c6da284d5f2286c8e3b2ea3cf1594c074
Successfully built sklearn
Installing collected packages: scipy, scikit-learn, sklearn
Successfully installed scikit-learn-0.20.3 scipy-1.2.1 sklearn-0.0
```

Instalar la librería matplotlib.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install matplotlib
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Collecting matplotlib
  Downloading https://files.pythonhosted.org/packages/f6/5d/a2b908620bfe5eeaaa06c85d297acadb912e813228ec3650eda54228fc76/matplotlib-2.2.4-cp27-cp27m-win32.whl (8.5MB)
[#####] 8.5MB 183kB/s
Collecting cycle>=0.10 (from matplotlib)
  Downloading https://files.pythonhosted.org/packages/f7/d2/e07d3ebb2bd7af696440ce7e754c59dd546ffe1bbe732c8ab68b9c834e61/cycler-0.10.0-py2.py3-none-any.whl
Collecting python-dateutil>=2.1 (from matplotlib)
  Using cached https://files.pythonhosted.org/packages/41/17/c62facbcbfd163c7f57f384d689e3a78baef1403648a6afb1d0866d87fbb/python_dateutil-2.8.0-py2.py3-none-any.whl
Collecting kiwisolver>=1.0.1 (from matplotlib)
  Downloading https://files.pythonhosted.org/packages/21/0b/23b4e5ffd2383354cdd4257b526bab807addf2896725b232eba81a03a2/kiwisolver-1.1.0-cp27-none-win32.whl (45kB)
[#####] 51kB 235kB/s
Requirement already satisfied: numpy>=1.7.1 in c:\users\fiorella\documents\reconocedor\env\lib\site-packages (from matplotlib) (1.16.4)
Collecting pyparsing>=2.0.4, <2.1.2, >=2.1.6, <2.0.1 (from matplotlib)
  Downloading https://files.pythonhosted.org/packages/dd/d9/3ec19e966301a6e25769976999bd7bbe552016f0d32b577dc9d63d2e0c49/pyparsing-2.4.0-py2.py3-none-any.whl (62kB)
[#####] 71kB 367kB/s
Collecting pytz (from matplotlib)
  Downloading https://files.pythonhosted.org/packages/3d/73/fe30c2daaaa0713420d0382b16fbb7b1409f532c56bdcc514bf7b6262bb6/pytz-2019.1-py2.py3-none-any.whl (510kB)
[#####] 512kB 499kB/s
Collecting backports.functiontools-lru-cache (from matplotlib)
  Downloading https://files.pythonhosted.org/packages/03/8e/2424c0e65c4a066e28f539364deee49b6451f8cfd4f718fefa50cc3dcf48/backports.functiontools_lru_cache-1.5-py2.py3-none-any.whl
Collecting six>=1.10 (from matplotlib)
  Using cached https://files.pythonhosted.org/packages/73/fb/00a976f728d0dfecfe898238ce23f502a721c0acbecfedb80e0d88c64e9/six-1.12.0-py2.py3-none-any.whl
Requirement already satisfied: setuptools in c:\users\fiorella\documents\reconocedor\env\lib\site-packages (from kiwisolver>=1.0.1->matplotlib) (41.0.1)
Installing collected packages: six, cycler, python-dateutil, kiwisolver, pyparsing, pytz, backports.functiontools-lru-cache, matplotlib
Successfully installed backports.functiontools-lru-cache-1.5 cycler-0.10.0 kiwisolver-1.1.0 matplotlib-2.2.4 pyparsing-2.4.0 python-dateutil-2.8.0 pytz-2019.1 six-1.12.0
```

Instalar Pandas

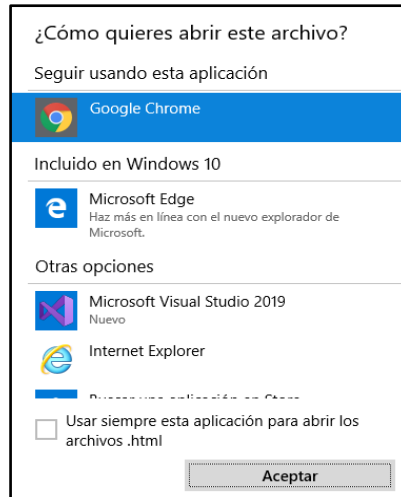
```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install pandas
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Collecting pandas
  Downloading https://files.pythonhosted.org/packages/34/cc/1911d56b9464de76f7ef34c6b1c66e82d3a394fd6d0925d5203e903d4ee/pandas-0.24.2-cp27-cp27m-win32.whl (7.2MB)
```

Instalar Jupyter.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>pip install jupyter
DEPRECATION: Python 2.7 will reach the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 won't be maintained after that date. A future version of pip will drop support for Python 2.7.
Collecting jupyter
  Using cached https://files.pythonhosted.org/packages/83/df/0f5dd132200728a86190397e1ea87cd76244e42d39ec5e88ef25b2abd7e/jupyter-1.0.0-py2.py3-none-any.whl
Collecting qtconsole (from jupyter)
  Downloading https://files.pythonhosted.org/packages/79/0b/efb5a694b6922bb85c35e4f1db6197daee23c764dd384023fc9517d79e26/qtconsole-4.5.1-py2.py3-none-any.whl (118kB)
[#####] 122kB 106kB/s
Collecting jupyter_console (from jupyter)
  Downloading https://files.pythonhosted.org/packages/77/82/6469cd7fccf7958cbe5dce2e623f1e3c5e27f1bb1ad36d90519bc2d5d370/jupyter_console-5.2.0-py2.py3-none-any.whl
Collecting ipynbkernel (from jupyter)
```

Iniciar Jupyter.

```
(env) C:\Users\FIORELLA\Documents\RECONOCEDOR>jupyter notebook
```



5.1.3 Componentes software

En la Figura 33 se muestra los componentes software del sistema.

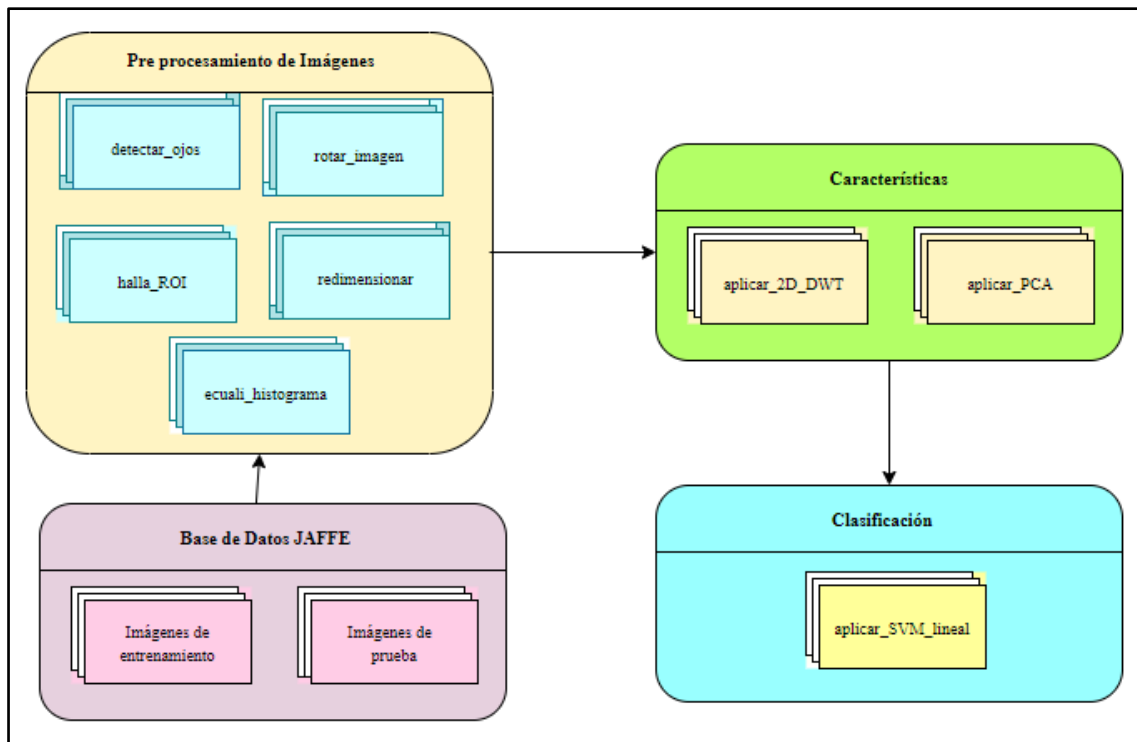


Figura 33: Componentes software

Fuente: (Github, 2019).

5.1.4 Pre-procesamiento de imágenes

Importamos las siguientes bibliotecas y librerías necesarias para el pre-procesamiento de imágenes: *OpenCV*, *numpy*, *math*, *os*, *dlib* e *imutils*.

En el sistema, se representa cada emoción con un valor entre cero y siete como se muestra en la Tabla 6.

Clase	Código
<i>Neutral</i>	0
<i>Felicidad</i>	1
<i>Enojo</i>	2
<i>Disgusto</i>	3

<i>Miedo</i>	4
<i>Tristeza</i>	5
<i>Sorpresa</i>	6

Tabla 6: Representación de clases

Fuente: (Github, 2019).

Para leer las imágenes en modo de escala de grises usamos el método `imread` de la biblioteca `OpenCV`. La biblioteca nos permite ingresar al sistema de archivos para leer las imágenes.

```
input_img = cv2.imread(dir_path + img, cv2.IMREAD_GRAYSCALE)
```

Se genera la Figura 34 con la librería `matplotlib`.



Figura 34: Imagen de la base de datos JAFFE leída usando `OpenCV` en `Python`

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

Identificamos los puntos de referencia faciales más importantes en las imágenes de expresiones mediante el predictor de la biblioteca `Dlib`.

Usamos el modelo predictor de la forma facial ya entrenado: `shape_predictor_68_face_landmarks.dat` (68 puntos de referencia facial) (Dlib).

```
detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor(landmarks_predictor_model)
```

```
shape = predictor(gray, rect)
shape = face_utils.shape_to_np(shape)
```

Se muestra los puntos de referencia faciales en la Figura 35.

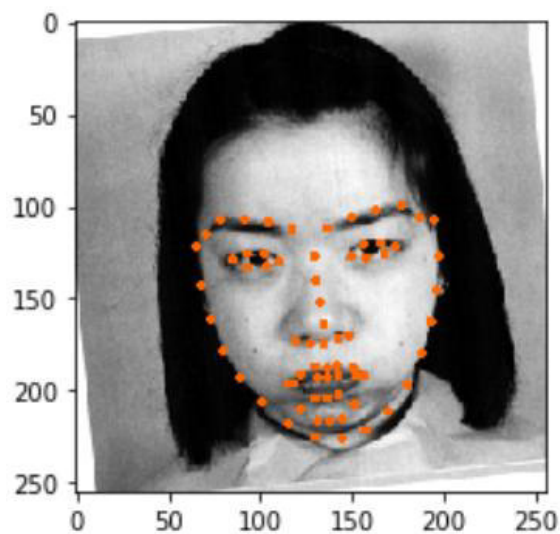


Figura 35: Puntos de referencia identificados en el rostro

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

Luego detectamos las coordenadas de los puntos centrales de los ojos izquierdo y derecho en la imagen para calcular la pendiente entre el eje X y la recta formada por los puntos centrales de los ojos. Se observa la detección de ojos derecho e izquierdo en la Figura 36.

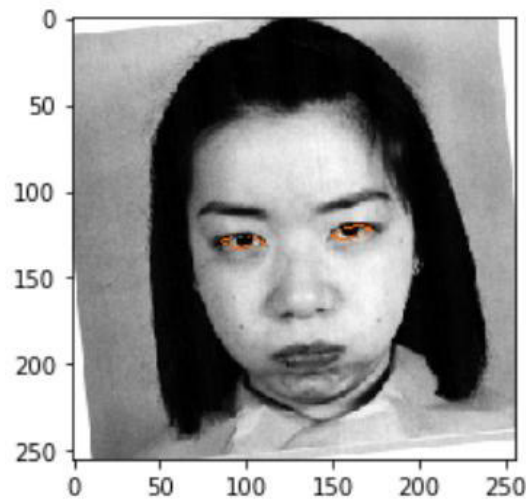


Figura 36: Detección de ojos en la imagen

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

Con estos datos hallamos la matriz de transformación que nos permitirá rotar la imagen hasta que la recta formada por los ojos tenga pendiente cero.

Para ubicar los puntos centrales de los ojos utilizamos los métodos *convexhull* y *moments* de la biblioteca *OpenCV* en *Python* y se muestra en la Figura 37.

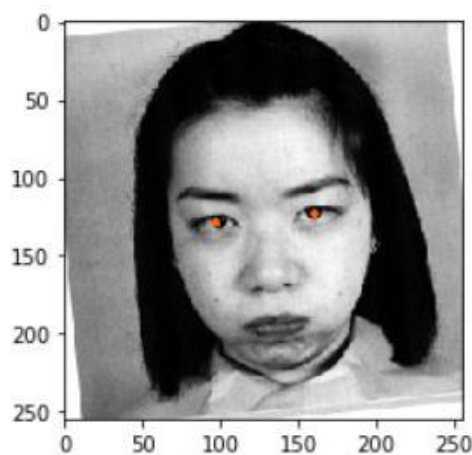


Figura 37: Localización de los puntos centrales de los ojos en el rostro

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

Para hallar la matriz de transformación y aplicar la matriz aplicamos respectivamente los métodos `getRotationMatrix2D` y `warpAffine` de la biblioteca OpenCV en Python.

```
rot_mat = cv2.getRotationMatrix2D(image_center, angle, 1.0)
result = cv2.warpAffine(image, rot_mat, image.shape[1::-1], flags=cv2.INTER_LINEAR)
```

Se muestra la rotación de la imagen en la Figura 38.

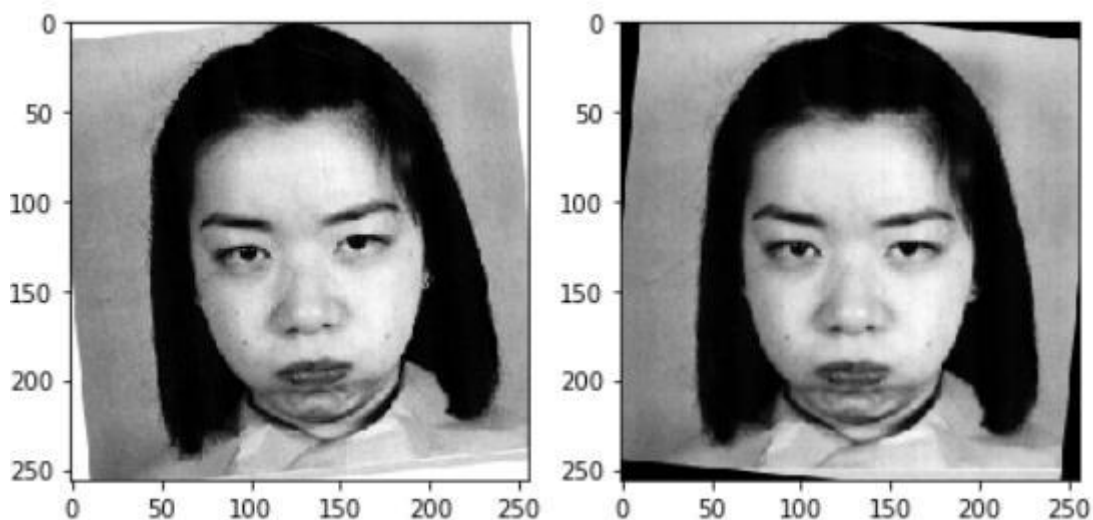


Figura 38: Rotación de la imagen

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

Localizamos la región de interés en la imagen a partir del punto medio de la recta que une a los puntos centrales de los ojos y se observa en la Figura 39.

Consecutivamente, se redimensiona la región de interés obtenida a 96x128. Para ello utilizamos el método `resize` de la biblioteca OpenCV y se visualiza en la Figura 40 y 41.

```
face_roi = cv2.resize(face_roi, (96,128))
```

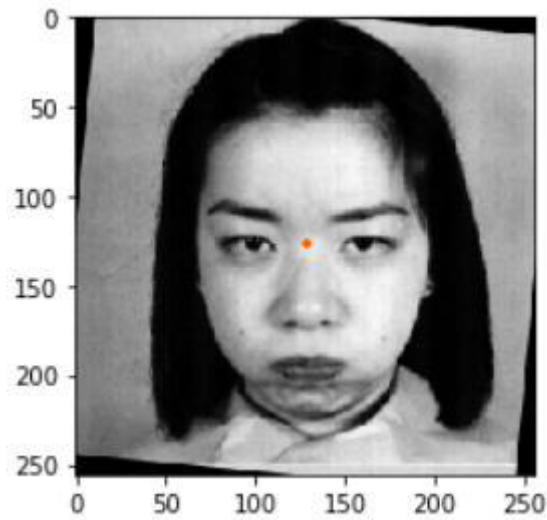


Figura 39: Localización del punto medio entre los ojos en la imagen

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

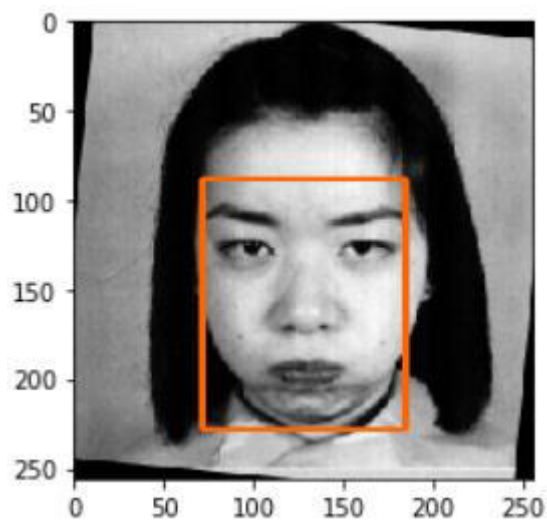


Figura 40: Región de interés(ROI)

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

Luego para ecualizar histogramas de la imagen usamos el método EqualizeHist de OpenCV.

```
face_roi = cv2.equalizeHist(face_roi)
```

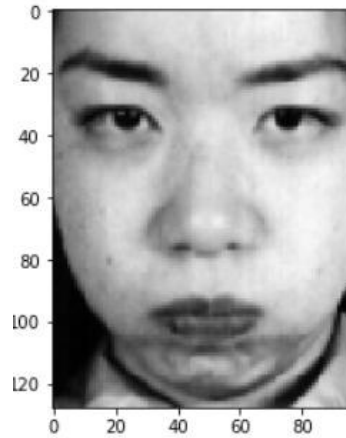


Figura 41: Imagen redimensionada

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

En la Figura 42, se observa la imagen después de aplicar ecualización de histograma.

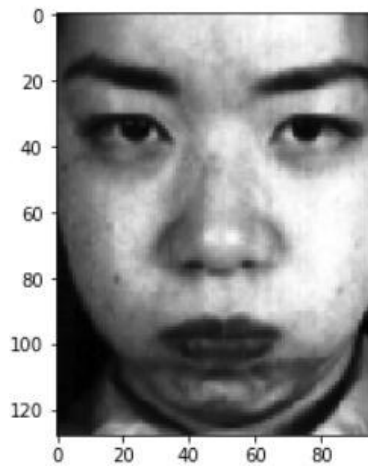


Figura 42: Imagen después de ecualizar su histograma

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

5.1.5 Extracción de características y reducción de dimensionalidad usando wavelets bidimensionales y PCA:

Importamos la librería pywt.

```
import pywt
```

Para aplicar la transformación de wavelet bidimensional usamos el método *dwt2* de la biblioteca *PyWavelets* en las imágenes previamente procesadas e indicamos el tipo de wavelet a utilizar:

- Wavelet : “bior1.3” (Wavelet simétrico y biortogonal) (Pywavelets, s.f.)

```
coeffs2 = pywt.dwt2(img, 'bior1.3')  
LL, (LH, HL, HH) = coeffs2
```

Con lo cual obtendremos la información de características en el coeficiente LL2 como se muestra en la Figura 43 y 44.

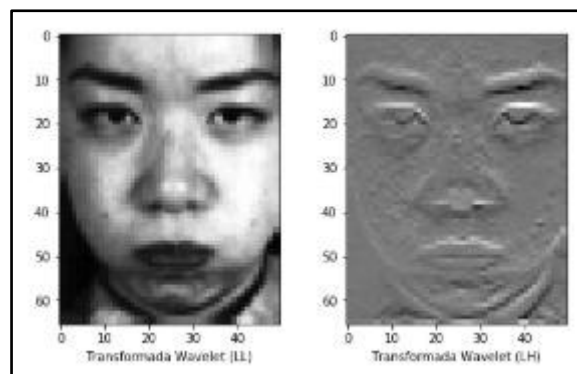


Figura 43: Coeficientes LL2 y LH2

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

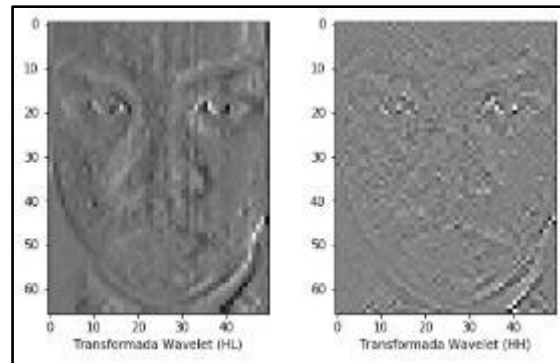


Figura 44: Coeficientes HL2 y HH2

Fuente de la imagen de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

Normalizamos el conjunto de datos antes de aplicar PCA, debido a que PCA asume que todos los datos tienen una distribución gaussiana o normal.

Para aplicar la normalización, se importa *StandardScaler* de la biblioteca *sklearn*. Con la finalidad de que cada característica se distribuya normalmente. De forma que se escale la distribución a una media de cero y una desviación estándar de uno.

Para escalar, aplicamos los métodos *fit* y *transform* consecutivamente.

```
from sklearn.preprocessing import StandardScaler
```

```
scaler = StandardScaler()
scaler.fit(X_new)
scaled_data = scaler.transform(X_new)
```

A continuación se proyectan los datos de las expresiones faciales de alta dimensión. Para ello importamos *PCA*, aplicamos el método *PCA* de *scikit learn* (Pedregosa, y otros, 2011) indicando como argumento la varianza deseada la cual será explicada por los componentes que se generarán.

Finalmente llamamos a los métodos *fit* y luego *transform* en el dataset de entrenamiento, con lo cual obtenemos los componentes principales que representan la dimensión inferior en la que se proyectan los datos de dimensión superior.

El dataset de entrenamiento se obtuvo mediante el método *train_test_split* de *scikit-learn* (Pedregosa, y otros, 2011).

```
from sklearn.decomposition import PCA
```

```
pca = PCA(n_components=0.99, whiten=False)
```

```
pca.fit(scaled_data)
```

```
X_tr, X_ts, y_tr, y_ts = train_test_split(scaled_data, Y, test_size=0.20, random_state=4)  
X_tr_pca = pca.transform(X_tr)
```

Como ejemplo, mostramos en la Figura 46 los datos de las expresiones faciales proyectados en un espacio 2D.

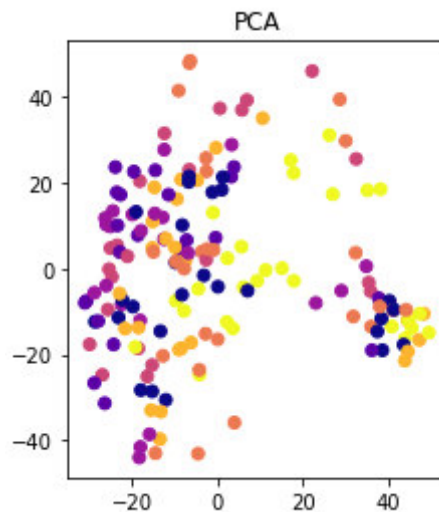


Figura 45: Datos proyectados en de dos componentes utilizando PCA

Fuente: (Github, 2019).

5.1.6 Configuración del clasificador SVM y entrenamiento:

Se utiliza la clase *GridSearchCV* de la biblioteca *scikit-learn* para seleccionar los parámetros de entrenamiento más apropiados para el modelo mediante una validación cruzada.

Configuramos los siguientes parámetros:

- Estimator: `SVC(kernel='linear')` (Modelo a evaluar)

- Param_grid: {"C": [0.1, 1, 10], "gamma": [1, 0.1, 0.01]} (Diccionario donde se indica los parámetros a evaluar como clave y el conjunto de elementos como valor)
- Scoring: 'accuracy'

Luego, GridSearchCV implementa el método "fit" utilizando los siguientes parámetros.

- x_tr_pca: Vector de entrenamiento.
- y_tr: Etiquetas objetivo para la clasificación del vector de entrenamiento.

```
param_grid = { "C" : [0.1, 1, 10] , "gamma" : [1, 0.1, 0.01]}
gs = GridSearchCV(estimator=SVC(kernel='linear'), param_grid=param_grid, scoring='accuracy', refit=True, verbose=2)
gs = gs.fit(X_tr_pca, y_tr)
```

Obtenemos como parámetros a utilizar los siguientes valores:

- 'C': 0.1
- 'gamma': 1

Luego se crea el clasificador de vectores de soporte (SVC) de la librería *scikit-learn* con los parámetros de la Tabla 7.

Parámetro	Valor	Descripción
C	0.1	Dureza del margen
Kernel	Linear	Tipo de núcleo o función kernel
Gamma	1	Coefficiente de kernel

Tabla 7: Parámetros del clasificador SVM

Fuente: (Github, 2019).

Para entrenar SVM pasamos como argumentos al método *fit*: Las características extraídas del conjunto de entrenamiento y sus respectivas etiquetas.

```
from sklearn.svm import SVC
```

```
bp = gs.best_params_
model = SVC(C=bp['C'], kernel='linear', gamma=bp['gamma'])
model.fit(X_tr_pca, y_tr)
```

Para la predicción se utiliza el método *predict* y le pasamos las características extraídas del dataset de prueba.

```
predictions = model.predict(X_ts_pca)
```

5.1.7 Evaluación

Se utiliza el 20% de imágenes del conjunto de datos JAFFE obtenido por el método *train_test_split* de la librería *scikit-learn*.

En la evaluación del sistema inteligente propuesto, se usan los métodos *classification_report* y *confusion_matrix* de la librería *metrics* para obtener el reporte de clasificación y la matriz de confusión respectivamente.

```
from sklearn.metrics import classification_report, confusion_matrix
```

5.2 Resultados experimentales

Los resultados experimentales obtenidos reflejan la efectividad y viabilidad del método propuesto.

Como se puede visualizar en la Figura 46, el sistema reconoció correctamente las emociones faciales de 40 imágenes, resaltadas en azul para diferenciarlas, de un total de 43 imágenes que conforman al conjunto de pruebas.

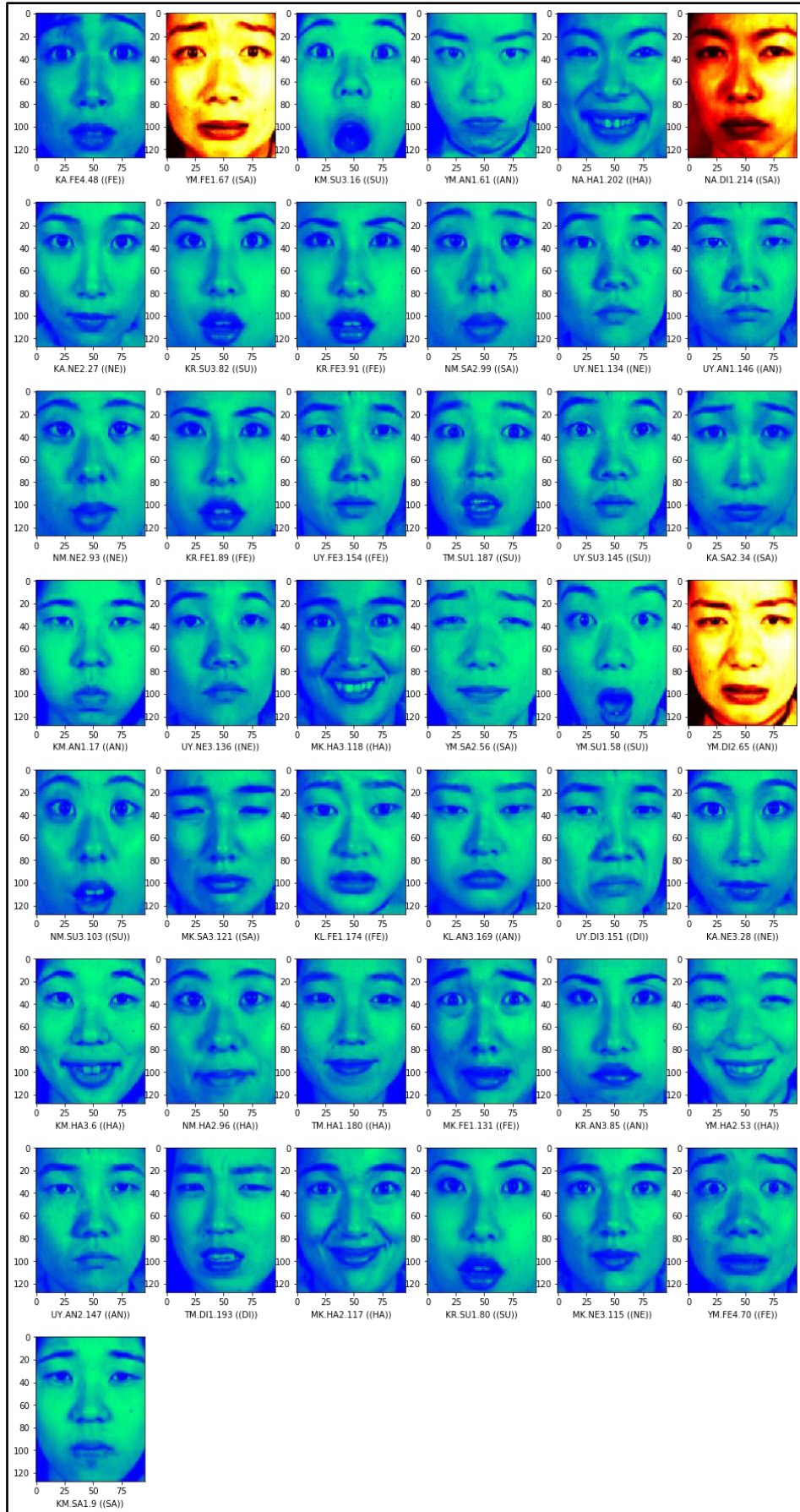


Figura 46: Resultados de la clasificación de emociones

Fuente de imágenes de JAFFE: (Lyons, Akemastu, Kamachi, & Gyoba, 1998).

La exactitud del sistema fue 93.02%.

Matriz de confusión

Para un mejor análisis del desempeño del sistema propuesto, se utiliza una matriz de confusión.

En las filas se representa el número de predicciones de cada clase (Es decir la emoción predicha) y en cada columna se representa a las instancias de clase real (Es decir la emoción real). En la Figura 48, se visualiza una matriz de confusión para una clasificación binaria (Hossin, Mohammad, N, & Sulaiman, 2015)..

	Actual Positive Class	Actual Negative Class
Predicted Positive Class	True positive (<i>tp</i>)	False negative (<i>fn</i>)
Predicted Negative Class	False positive (<i>fp</i>)	True negative (<i>tn</i>)

Figura 47: Matriz de confusión para una clasificación binaria

Fuente: (Hossin, Mohammad, N, & Sulaiman, 2015).

Donde, un verdadero positivo es un resultado donde el modelo predice correctamente la clase positiva.

Un verdadero negativo es un resultado en el que el modelo predice correctamente la clase negativa. Mientras que un falso positivo es un resultado donde el modelo predijo de manera incorrecta a la clase positiva y un falso negativo es un resultado donde el modelo predijo de manera incorrecta a la clase negativa.

Los valores de las diagonales expresan el número el número de imágenes clasificadas correctamente.

La confusión se representa por el número de imágenes clasificadas incorrectamente (Los elementos fuera de la diagonal), ya que se confundieron con otra clase.

La Tabla 8, muestra la matriz de confusión del modelo propuesto.

Clase real	Clase Predicha						
	Neutro	Feliz	Enojado	Disgustado	Asustado	Triste	Sorprendido
Neutro	6	0	0	0	0	0	0
Feliz	0	7	0	0	0	0	0
Enojado	0	0	6	0	0	0	0
Disgustado	0	0	1	2	0	1	0
Asustado	0	0	0	0	7	1	0
Triste	0	0	0	1	0	5	0
Sorprendido	0	0	0	0	0	0	7

Tabla 8: Matriz de confusión

Fuente: Elaboración propia.

En la Figura 48, se visualiza gráficamente la matriz de confusión no normalizada obtenida.

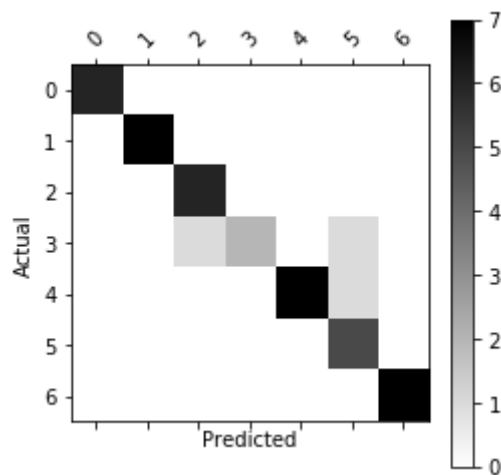


Figura 48: Matriz de confusión sin normalización, elaborado en Python

Fuente: Elaboración propia.

La matriz de confusión normalizada es útil para tener una mejor visualización de cómo se predicen las etiquetas.

En la tabla 9, se muestra la matriz de confusión normalizada.

Clasereal	Clase predicha						
	Neutro	Feliz	Enojado	Disgustado	Asustado	Triste	Sorprendido
Neutro	1	0	0	0	0	0	0
Feliz	0	1	0	0	0	0	0
Enojado	0	0	1	0	0	0	0
Disgustado	0	0	0.25	0.5	0.25	0	0
Asustado	0	0	0	0	0.875	0.125	0
Triste	0	0	0	0.167	0	0.833	0
Sorprendido	0	0	0	0	0	0	1

Tabla 9: Matriz de confusión normalizada

Fuente: Elaboración propia.

En la Figura 49, se observa la matriz de confusión obtenida normalizada.

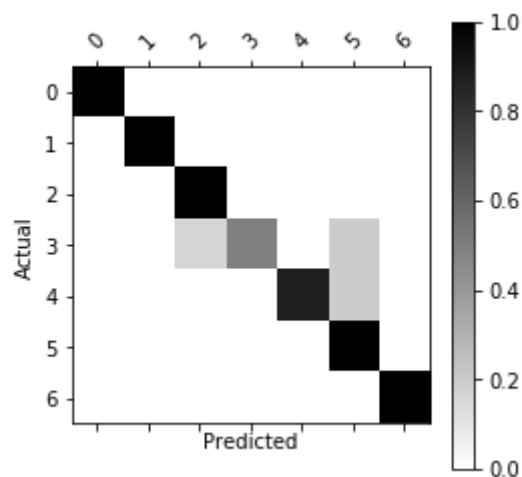


Figura 49: Matriz de confusión normalizada, elaborado en Python

Fuente: Elaboración propia.

Reporte de clasificación

El reporte de clasificación muestra las principales métricas de clasificación por cada clase: Precisión, sensibilidad, puntaje de F1 y soporte del modelo. Por lo tanto brinda mayor visibilidad del comportamiento del clasificador multiclase.

En el reporte de clasificación, se observa que el sistema tiene mayor precisión en las emociones: Felicidad, sorpresa y neutral. Y menor precisión en la emoción del disgusto.

Las emociones con mayor sensibilidad fueron: Neutral, felicidad, enojo, tristeza y sorpresa.

En la Tabla 10, se observa el reporte de clasificación.

	Precisión	Sensibilidad	Puntaje F1	Soporte
Neutral	1.00	1.00	1.00	6
Felicidad	1.00	1.00	1.00	7
Enojo	0.86	1.00	0.92	6
Disgusto	1.00	0.50	0.67	4
Miedo	1.00	0.88	0.93	8
Tristeza	0.71	1.00	0.83	5
Sorpresa	1.00	1.00	1.00	7
Micro Promedio	0.93	0.93	0.93	43
Macro Promedio	0.94	0.91	0.91	43
Promedio ponderado	0.95	0.93	0.93	43

Tabla 10: Reporte de clasificación

Fuente: Elaboración propia.

En la Tabla 11 se muestra los tiempos de ejecución obtenidos en cada etapa del método propuesto.

Ítem	Etapas	Tiempo(Segundos)
1	Pre-procesamiento	19.459
2	Extracción de características usando 2D-DWT	0.158
3	Entrenamiento sobre el conjunto de entrenamiento	0.013
4	Clasificación sobre el conjunto de prueba	0.003

Tabla 11: Tiempos obtenidos por etapas del método propuesto

Fuente: Elaboración propia.

En la tabla 12, se visualiza la comparación de precisiones obtenidas al aplicar el método propuesto usando imágenes de JAFFE y CK+. Se utilizaron 354 imágenes de la base de datos CK+ para entrenar y validar el sistema.

Ítem	Dataset	Método	Precisión(%)
1	JAFFE	2D DWT+PCA+SVM	93.02
2	CK+	2D DWT+PCA+SVM	92.95

Tabla 12: Comparación de precisiones en JAFFE y CK+

Fuente: Elaboración propia.

A modo de comparación se probó el sistema inteligente sin realizar la ecualización de histogramas en las imágenes durante el pre procesamiento. Los resultados obtenidos en precisión, matriz de confusión y reporte de clasificación fueron los mismos que cuando se aplicó la ecualización en el conjunto de datos.

5.3 Comparación con otros métodos

En la Tabla 13 se observa una comparación entre la precisión obtenida por los métodos del estado del arte y el sistema propuesto en el presente trabajo sobre imágenes de la base de datos JAFFE.

Precisiones en base de datos JAFFE:

Ítem	Artículo	Método	Precisión(%)
1	(Wei, Jia, Feng, Chen, & Chu, 2019)	CNN+C4.5+Bosque Aleatorio	94.41
2	(Wang, Li, Song, & Rong, 2019)	CNN+SVM	98.9
3	(Chang & Ryoo, 2017).	PCA+LDA+SVM	81.5
4	(Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019)	CLAHE+DWT+CNN	98.63
5	Propuesto	2D DWT+PCA+SVM	93.02

Tabla 13: Comparación entre métodos

Fuente: Elaboración propia.

En la tabla 13, se observa que el sistema inteligente propuesto superó en precisión al método propuesto en el artículo (Chang & Ryoo, 2017). Sin embargo; no superó en precisión a los métodos de: (Wei, Jia, Feng, Chen, & Chu, 2019), (Wang, Li, Song, & Rong, 2019) y (Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019). Estas técnicas tuvieron en común el uso de redes neuronales convolucionales(CNN).

Por ello se decidió cambiar solo el clasificador SVM por CNN en el método propuesto con la finalidad de medir la variación en la precisión. De tal manera que se usó una red neuronal convolucional para entrenar y clasificar.

Se implementó un modelo tipo secuencial de red neuronal convolucional (Analytics Vidhya, s.f.) usando la librería *Keras* (Chollet & Francois, 2015). Las capas son las siguientes:

1. La primera capa es una convolucional con 25 nodos, recibe las características de la imagen. La forma de la imagen que recibe es 66x50 en escala de grises. El tamaño de la matriz filtro para la convolución es de 3x3 y la función de activación en esta capa es ReLu(Activación lineal rectificada).
2. La segunda capa reduce la muestra tomando el máximo valor sobre la ventana definida en el parámetro pool_size en este caso su valor es 2x2. En otras palabras reduce la dimensionalidad reduciendo el número de píxeles, al final se obtiene una nueva representación de la imagen.
3. La tercera capa flatten funciona como conexión entre las capas Conv2D y la capa densa.
4. La cuarta capa es una capa densa con 100 nodos y función de activación ReLu.
5. La última capa de salida es densa con 7 nodos, uno para cada posible salida (0-6). Y función de activación softmax. Softmax hace que la salida sume uno, por lo que la salida puede interpretarse como probabilidades. El modelo hace la predicción según la opción que tiene mayor probabilidad.

Se visualiza en la Figura 51, la arquitectura de la red neuronal convolucional implementada.

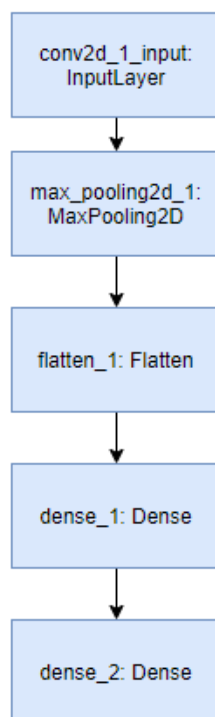


Figura 50: Arquitectura del CNN implementado, basado en (Analytics Vidhya, s.f.)

Fuente: (Analytics Vidhya, s.f.).

Luego se aplicó la técnica “Data Augmentación”, la cual nos permitió aumentar el tamaño y diversidad del dataset por medio de alteraciones en el conjunto de datos original. El nuevo conjunto de datos está conformado por 4890 imágenes.

En la Figura 51, se observa que se alcanzó una precisión de 98.67%.El tiempo de entrenamientofue 412 segundos.

```
Epoch 15/30
31/31 [=====] - 10s 318ms/step - accuracy: 1.0000 - loss: 0.0067 - val_accuracy: 0.9867 - val_loss: 0.0501
Epoch 16/30
31/31 [=====] - 12s 395ms/step - accuracy: 1.0000 - loss: 0.0056 - val_accuracy: 0.9867 - val_loss: 0.0486
Epoch 17/30
31/31 [=====] - 18s 588ms/step - accuracy: 1.0000 - loss: 0.0048 - val_accuracy: 0.9867 - val_loss: 0.0499
Epoch 18/30
31/31 [=====] - 16s 501ms/step - accuracy: 1.0000 - loss: 0.0042 - val_accuracy: 0.9877 - val_loss: 0.0477
Epoch 19/30
31/31 [=====] - 10s 311ms/step - accuracy: 1.0000 - loss: 0.0038 - val_accuracy: 0.9867 - val_loss: 0.0484
Epoch 20/30
31/31 [=====] - 10s 314ms/step - accuracy: 1.0000 - loss: 0.0033 - val_accuracy: 0.9857 - val_loss: 0.0466
Epoch 21/30
31/31 [=====] - 10s 315ms/step - accuracy: 1.0000 - loss: 0.0030 - val_accuracy: 0.9857 - val_loss:
```

Figura 51: Resultados usando CNN sobre el conjunto de validación

Fuente: Elaboración propia.

En la Figura 52, se muestran las líneas de aprendizaje de pérdida y precisión para entrenar y probar el sistema.

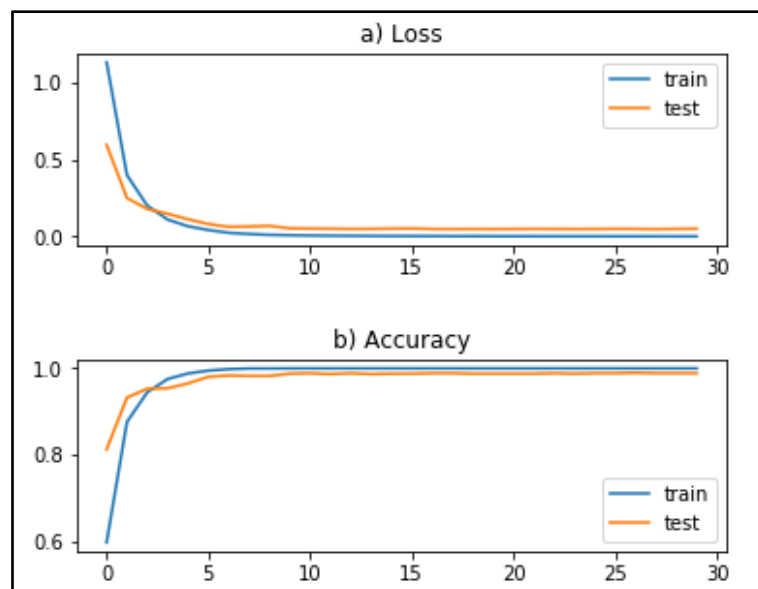


Figura 52: Línea de aprendizaje de pérdida y precisión

Fuente: Elaboración propia

En la Tabla 14, se visualiza la comparación de precisiones entre el método propuesto usando clasificador SVM lineal y empleando un clasificador CNN.

Ítem	Método	Precisión(%)
1	2D DWT+PCA+CNN	98.67
2	2D DWT+PCA+SVM	93.02

Tabla 14: Comparación de precisión después de variar el clasificador

Fuente: Elaboración propia.

La precisión obtenida por la variación del método propuesto usando CNN superó en esta prueba a la precisión obtenida usando clasificador SVM sobre el conjunto de datos JAFFE.

Capítulo VI Discusión, Conclusiones y recomendaciones

En el presente capítulo se presenta la discusión, conclusiones y recomendaciones.

6.1 Discusión

- El sistema propuesto en el presente trabajo demuestra una alta precisión (93.02%) en el reconocimiento de las emociones faciales debido al uso de Wavelets bidimensionales y algoritmo PCA. Los cuales permitieron extraer información relevante de las imágenes y que fue de vital importancia para la clasificación de la emoción.
- En comparación con los métodos de (Wang, Li, Song, & Rong, 2019), (Wei, Jia, Feng, Chen, & Chu, 2019), (Happy, Dantcheva, & Bremond, 2019) y (Ilyas, Beladgham, Merit, & Taleb-Ahmed, 2019) basados en redes neuronales convolucionales; el método propuesto no requirió de una extensa cantidad de imágenes para entrenar al sistema ni tampoco de configuraciones complejas. Sin embargo, dichos sistemas basados en redes neuronales reflejan una precisión promedio entre 90.1% y 99.9%. La precisión obtenida en este trabajo también se encuentra dentro de dicho rango.
- El modelo evidenció muy buenos resultados para reconocer las emociones de sorpresa (1.0), felicidad (1.0) y neutral (1.0) según lo observado en la métrica puntaje F1 del reporte de clasificación. Posiblemente por la marcada diferencia de las expresiones faciales asociadas a estas emociones comparada con las expresiones faciales asociadas a las demás emociones.
- Para medir el impacto de la técnica de ecualización de histogramas, se probó al sistema cuando aplicaba esta en las imágenes y cuando no lo aplicaba. Como resultado no se observó variación en la precisión ni en otros indicadores como la matriz de confusión o el reporte de clasificación. Debido a que las imágenes de JAFFE fueron capturadas en entornos de iluminación controlados.

- Para medir la influencia del método de clasificación seleccionado, se probó el sistema cambiando solo el clasificador a redes neuronales convolucionales (Analytics Vidhya, s.f.) y aumentando el tamaño del conjunto de datos. Se alcanzó mayor precisión en el reconocimiento. Como es bien conocido, las redes neuronales incrementan su precisión cuando se aplican sobre grandes volúmenes de datos.

6.2 Conclusiones y recomendaciones

- Al término de la presente tesis se puede concluir que el sistema inteligente, basado en Wavelets bidimensionales y Máquinas de Vectores de Soporte, es efectivo para reconocer emociones faciales en imágenes digitales.
- Se obtuvo una precisión de 93.02% al clasificar imágenes de la base de datos JAFFE y 92.95% cuando se aplicó sobre imágenes de CK+. Estos valores deben tomarse en cuenta cuando se desee realizar cualquier implementación en el futuro.

Asimismo, se puede afirmar la alta efectividad del sistema basado en Wavelets bidimensionales y redes neuronales convolucionales para el reconocimiento automático de emociones faciales, ya que se obtuvo experimentalmente una precisión de 98.67% sobre imágenes de la base de datos JAFFE.

- El sistema inteligente desarrollado distinguió entre 7 tipos de emociones diferentes basándose en la clasificación planteada por Paul Ekman: enojo, desagrado, miedo, alegría, tristeza y sorpresa. Y adicionalmente se consideró la emoción neutral presente en la base de datos JAFFE. Sin embargo, las emociones humanas son por naturaleza más diversas, complejas, y dinámicas. Es por ello, que la elaboración de un sistema que identifique un mayor número de emociones puede ser considerado para futuros trabajos.

REFERENCIAS BIBLIOGRÁFICAS

- Ai-Sumaidae , S. A., Dlay, S. S., Woo, W. L., & Chambers, J. A. (2015). Facial expression recognition using local Gabor gradient code-horizontal diagonal descriptor. *2nd IET International Conference on Intelligent Signal Processing 2015 (ISP)*.
- Allegreta, I., Marangoni, B., Manzari, P., Porfido, C., Terzano, R., De Pascale, O., & Senesi, G. (2020). Macro-classification of meteorites by portable energy dispersive X-ray fluorescence.
- Amari, S., & Wu, S. (1999). Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*.
- Analytics Vidhya*. (s.f.). Obtenido de <https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets/>
- Bradski, & G. (2000). The OpenCV Library. *Journal of Software Tools*.
- Buitinck, L., Louppe, G., Pedregosa, F., Mueller, A., Olivier, G., Niculae, V., . . . Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. *European Conference on Machine Learning and Principles and Practices of Knowledge Discovery in Databases*. Obtenido de https://scikit-learn.org/stable/auto_examples/decomposition/plot_pca_vs_lda.html
- Challenges in representation learning*. (s.f.). Obtenido de <https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge>
- Chang, J., & Ryoo, S. (2017). Implementation of an improved facial emotion retrieval method in multimedia system. *Multimed Tools Appl*.
- Chollet, & Francois. (2015). *Github*. Obtenido de <https://github.com/fchollet/keras>
- Connolly, H., Lefevre, C., Young, A., & Lewis, G. (2019). Emotion recognition ability: Evidence for a supramodal factor and its links to social cognition. *Cognition*.
- Cunha Santiago, H. D., Ren Tsang, I., & Cavalcanti, G. D. (2016). Facial expression Recognition based on Motion Estimation. *International Joint Conference on Neural Networks (IJCNN)*.
- Darwin, C. (1872). *The Expression of the Emotions in Man and Animals*.
- Díaz, N., Barredo, A., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., . . . Herrera, F. (2019). Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI. *Information Fusion*.
- Ding, H., Zhou, S. K., & Chellappa, R. (2017). FaceNet2ExpNet: Regularizing a Deep Face Recognition Net for Expression Recognition. *12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*.

- Dlib. (s.f.). Obtenido de http://dlib.net/files/shape_predictor_68_face_landmarks.dat.bz2
- Ekman, P., Friesen, W., & Hager, J. (2002). Facial Action Coding System - The Manual on CD-ROM [CD-ROM].
- Ekman, P., Sorenson, R., & Friesen, W. (1969). Pan-cultural elements in facial displays of emotion.
- Erevelles, S., Shuba, S., & Rangel, S. (2003). Consumer Satisfaction for Internet Service Providers: An Analysis of Underlying Processes. *Information Technology and Management*.
- Española, R. A. (s.f.). *Diccionario de la lengua española*. Obtenido de <https://dle.rae.es/emoción>
- Fisher, R. A., D, S., & S, F. R. (1936). THE USE OF MULTIPLE MEASUREMENTS IN TAXONOMIC PROBLEMS. En *Annals of human genetics*.
- Github. (24 de March de 2019). Obtenido de <https://github.com/anas-899/facial-expression-recognition-Jaffe>
- Gove, R., & Faytong, J. (2012). Machine Learning and Event-Based Software Testing: Classifiers for Identifying Infeasible GUI Event Sequences. En *Advances in Computers*.
- Guru, D. S., Suraj, M. G., & Manjunath, S. (2010). Fusion of covariance matrices of PCA and FLD. *ELSEVIER*.
- Hamester, D., Barros, P., & Wermter, S. (2015). Face expression recognition with a 2-channel Convolutional Neural Network. *International Joint Conference on Neural Networks (IJCNN)*.
- Happy, S. L., & Routray, A. (2015). Automatic facial expression recognition using features of salient facial patches. *IEEE Transactions on Affective Computing*.
- Happy, S., Dantcheva, A., & Bremond, F. (2019). A Weakly Supervised learning technique for classifying facial expressions. *Pattern Recognition Letters*.
- Hossin, Mohammad , N, M., & Sulaiman. (2015). A Review on Evaluation Metrics for Data Classification Evaluations. *International Journal of Data Mining & Knowledge Management Process*.
- Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* .
- Hurtado, J. E. (2003). An examination of methods for approximating implicit limit state functions from the viewpoint of statistical learning theory. *Structural Safety*.
- Iglesias, S., Del Castillo, A., & Muñoz, J. (2016). Reconocimiento facial de expresión emocional: diferencias por licenciaturas. *Acta de Investigación Psicológica*.

- Ilyas, R., Beladgham, M., Merit, K., & Taleb-Ahmed, A. (2019). Improved Facial Expression Recognition Based on DWT Feature for Deep CNN. *Electronics*.
- Jung, H., Lee, S., Yim, J., Park, S., & Kim, J. (2015). Joint Fine-Tuning in Deep Neural Networks for Facial Expression Recognition. *IEEE International Conference on Computer Vision (ICCV)*.
- Kanade, T., Cohn, J. F., & Tian, Y. (2000). Comprehensive Database for Facial Expression Analysis. *Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*.
- Khan, G., Samyan, S., Ghani, M., Shahid, M., & Wahla, S. (2019). A survey on analysis of human faces and facial expressions datasets. *International Journal of Machine Learning and Cybernetics*.
- Khatchadourian, R. (12 de 01 de 2015). Obtenido de The New Yorker: <https://www.newyorker.com/magazine/2015/01/19/know-feel>
- King, D. E. (2009). Dlib-ml: A Machine Learning Toolkit. *The Journal of Machine Learning Research*.
- Kumar, D., Kumar, P., Tripathi, S., & Kumar, S. (2019). Performance evaluation of classification methods with PCA and PSO for diabetes. *Network Modeling Analysis in Health Informatics and Bioinformatics*.
- Lee, G. R., Gommers, R., Waselewski, F., Wohlfahrt, K., & O'Leary, A. (2019). PyWavelets: A Python package for wavelet analysis. *The Journal of Open Source Software*.
- Li, D., Zhang, C., Liu, H., Su, J., Tan, X., Liu, Q., & Liao, G. (2019). A Fast Cross-Range Scaling Algorithm for ISAR Images Based on the 2-D Discrete Wavelet Transform and Pseudopolar Fourier Transform. *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*.
- Li, J., Wang, W., Deng, N., & Xin, B. (2019). A Novel Digital Method for Weave Pattern Recognition Based on Photometric Differential Analysis. *Measurement*.
- Li, S., Deng, W., & Du, J. (2017). Reliable Crowdsourcing and Deep Locality-Preserving Learning for Expression Recognition in the Wild. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Lisungu, O. T., & Sumbwanyambe, M. (2019). A Selective Image Encryption Scheme Based on 2D DWT, Henon map and 4D Qi Hyper-chaos. *Acceso IEEE*.
- Liu, M., Li, S., Shan, S., & Chen, X. (2015). AU-inspired Deep Networks for Facial Expression Feature Learning. *Neurocomputing*.
- Liu, M., Shan, S., Wang, R., & Chen, X. (2014). Learning Expressionlets on Spatio-temporal Manifold for Dynamic Facial Expression Recognition. *IEEE Conference on Computer Vision and Pattern Recognition*.

- Liu, P., Han, S., Meng, Z., & Tong, Y. (2014). Facial Expression Recognition via a Boosted Deep Belief Network. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., & Ambadar, Z. (2010). The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*.
- Lyons, M. J., Akemastu, S., Kamachi, M., & Gyoba, J. (1998). Coding Facial Expressions with Gabor Wavelets. *Third IEEE International Conference on Automatic Face and Gesture Recognition*, (págs. 200-205).
- Lyons, M., Akamatsu, S., Kamachi, M., & Gyoba, J. (1998). Coding Facial Expressions with Gabor Wavelets. *Proceedings of the 3rd. International Conference on face & Gesture Recognition*.
- Ma, W., Zhang, X., Xin, Y., & Li, S. (2019). Study on short-term network forecasting based on SVM-MFA algorithm. *ELSEVIER*.
- Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.
- McKinney, W. (2010). Data Structures for Statistical Computing in Python. *PROC. OF THE 9th PYTHON IN SCIENCE CONF.*
- Meng, Z., Liu, P., Cai, J., Han, S., & Tong, Y. (2017). Identity-Aware Convolutional Neural Network for Facial Expression Recognition. *12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*.
- Ming-wei, G., Yu-zhou, Z., Jun-ping, X., Chen-bin, Z., & Zong-hai, C. (2014). Review of object detection methods based on SVM. *Control and Decision*.
- Minh Vo, D., Sugimoto, A., & Hoang Le, T. (2016). Facial expression recognition by re-ranking with global and local generic features. *23rd International Conference on Pattern Recognition (ICPR)*.
- Mollahosseini, A., Chan, D., & Mahoor, M. H. (2016). Going deeper in facial expression recognition using deep neural networks. *IEEE Winter Conference on Applications of Computer Vision (WACV)*.
- Navarro, J. (2018). *Detección de fallas y selección de variables en un proceso multivariado usando análisis de señales y sistemas inteligentes*. Corporación Mexicana de Investigación en Materiales S.A. de C.V, México.
- Navnidhi, C., & Basha2, S. J. (2012). Comparison of Digital Image watermarking Methods DWT & DWT-DCT on the Basis of PSNR. *International Journal of Innovative Research in Science, Engineering and Technology*.

- Nwosu, L., Wang, H., Unwala, I., Yang, X., & Zhang, T. (2017). Deep Convolutional Neural Network for Facial Expression Recognition Using Facial Parts. *IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress*.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12.
- Pywavelets. (s.f.). *Wavelet Browser*. Obtenido de <http://wavelets.pybytes.com/wavelet/bior1.3/>
- Ren, F., & Quan, C. (2012). Linguistic-based emotion analysis and recognition for measuring. *Information Technology and Managment*.
- Rossum, V., & G. (1995). Python reference manual. *Centrum voor Wiskunde en Informatica Amsterdam*.
- Shan, C., Gong, S., & McOwan, P. W. (2009). Facial expression recognition based on Local Binary Patterns: A comprehensive study. *Image and Vision Computing*.
- Shih, F. Y., Wang, P. S.-P., & Chuang, C.-F. (2008). Performance Comparisons of Facial Expression Recognition in Jaffe Database. *International Journal of Pattern Recognition and Artificial Intelligence*.
- Sikka, K., Sharma, G., & Bartlett, M. (2016). LOMo: Latent Ordinal Model for Facial Analysis in Videos. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Teixeira Lopes, A., De Aguiar, E., De Souza, A. F., & Oliveira-Santos, T. (2017). Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order. *Pattern Recognition*.
- Van der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering*.
- Wang, Y., Li, Y., Song, Y., & Rong, X. (2019). Facial Expression Recognition Based on Random Forest and Convolutional Neural Network. *Information*.
- Wei, W., Jia, Q., Feng, Y., Chen, G., & Chu, M. (2019). Multi-modal facial expression feature based on deep-neural networks. *Journal on Multimodal User Interfaces*.
- Wen, G., Hou, Z., Li, D., Jiang, L., & Xun, E. (2017). Ensemble of Deep Neural Networks with Probability-Based Fusion for Facial Expression Recognition. *Cognitive Computation*.
- Zhao, X., Liang, X., Liu, L., Li, T., Han, Y., Vasconcelos, N., & Yan, S. (2016). Peak-Piloted Deep Network for Facial Expression Recognition. *Computer Vision - ECCV*.

- Zhiyong, Z., Mugang, L., & Shengji, Z. (2005). Short-Term Load Forecasting Based on Wavelet Transform and Support Vector Machine. *Microcomputer applications*.
- Zhou, L., Wang, H., Lin, S., Hao, S., & Lu, Z.-M. (2019). Face recognition based on local binary pattern and improved Pairwise-constrained Multiple Metric Learning. *Multimedia Tools and Applications*.