



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO



Instituto Tecnológico de León

Diseño evolutivo de descriptores para la caracterización y reconocimiento de imágenes de expresiones faciales

Tesis

Que presenta:

Ing. Manuel Alejandro Torres Fonseca

**Para obtener el grado de
Maestro en Ciencias de la Computación**

Con la Dirección de:

Dr. Manuel Ornelas Rodríguez

Y Co Dirección de:

Dr. Valentín Calzada Ledesma

Revisores

Dr. Alfonso Rojas Domínguez

Dr. Juan Martín Carpio Valadez

enero de 2021



León, Gto. 10/enero/2022

C. MANUEL ALEJANDRO TORRES FONSECA PRESENTE

De acuerdo al fallo emitido por el Comité Tutorial, integrado por los: C. Dr. Manuel Ornelas Rodríguez, Dr. Valentín Calzada Ledezma, Dr. Juan Martín Carpio Valadez, Dr. Alfonso Rojas Domínguez y considerando que llena todos los requisitos en los Lineamientos Generales para la Operación del Posgrado del Tecnológico Nacional de México, se autoriza la impresión del trabajo de tesis titulado: "Diseño Evolutivo de Descriptores para la caracterización y reconocimiento de imágenes faciales".

Lo que hacemos de su conocimiento para los efectos y fines correspondientes.

ATENTAMENTE

Excelencia en Educación Tecnológica® (Ciencia Tecnología y Libertad)

DAVID ASAFEL GUTIÉRREZ HERNÁNDEZ
JEFE DE LA DEPI

ccp. Archivo



Av. Tecnológico s/n Fracc. Industrial
Julián de Obregón C.P 37290
León, Gto. México Tel. 01 (477) 7105200,
e-mail: tecleon@leon.tecnm.mx
tecnm.mx | leon.tecnm.mx





León, Guanajuato., a 04 de enero del 2022

C. ING. LUIS ROBERTO GALLEGOS MUÑOZ
JEFE DE SERVICIOS ESCOLARES
PRESENTE

Por este medio hacemos de su conocimiento que la tesis titulada "Diseño evolutivo de descriptores para la caracterización y reconocimiento de imágenes de expresiones faciales", ha sido leída y aprobada por los miembros del Comité Tutorial para su evaluación por el jurado del acto de examen de grado al alumno (a) C. Manuel Alejandro Torres Fonseca, con número de control M14240803 como parte de los requisitos para obtener el grado de Maestro(a) en Ciencias de la Computación (MCCOM-2011-05).

Sin otro particular por el momento, quedamos de Usted.

ATENTAMENTE
COMITÉ TUTORIAL


Dr. Manuel Ornelas Rodríguez.
DIRECTOR


Dr. Valentín Calzada Ledesma
CODIRECTOR


Dr. Alfonso Rojas Domínguez
REVISOR


Dr. Juan Martín Carpio Valadez
REVISOR



DECLARACION DE AUTENTICIDAD Y DE NO PLAGIO

Yo, **Manuel Alejandro Torres Fonseca** identificado con No. Control **M14240803**, alumno del programa de la **Maestría en Ciencias de la Computación**, autor de la Tesis titulada: "**Diseño evolutivo de descriptores para la caracterización y reconocimiento de imágenes de expresiones faciales**" DECLARO QUE:

1.- El presente trabajo de investigación, tema de la tesis presentada para la obtención del título de **MAESTRO EN CIENCIAS DE LA COMPUTACIÓN** es original, siendo resultado de mi trabajo personal, el cual no he copiado de otro trabajo de investigación, ni utilizado ideas, fórmulas, ni citas completas "stricto sensu", así como ilustraciones, fotografías u otros materiales audiovisuales, obtenidas de cualquier tesis, obra, artículo, memoria, etc. en su versión digital o impresa.

2.- Declaro que el trabajo de investigación que pongo a consideración para evaluación no ha sido presentado anteriormente para obtener algún grado académico o título, ni ha sido publicado en sitio alguno.

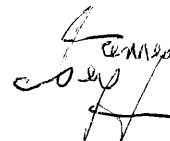
3.- Declaro que las pruebas o experimentos derivados de esta investigación fueron realizados bajo el consentimiento de los involucrados y con fines estrictamente académicos conforme a criterios éticos de confidencialidad.

Soy consciente de que el hecho de no respetar los derechos de autor y hacer plagio, es objeto de sanciones universitarias y/o legales por lo que asumo cualquier responsabilidad que pudiera derivarse de irregularidades de la tesis, así como de los derechos sobre la obra presentada.

Asimismo, me hago responsable ante el Tecnológico Nacional de México/Instituto Tecnológico de León o terceros, de cualquier irregularidad o daño que pudiera ocasionar por el incumplimiento de lo declarado.

De identificarse falsificación, plagio, fraude, o que el trabajo de investigación haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas pecuniarias o legales que se deriven de ello sometiéndome a las normas establecidas en los Lineamientos y Disposiciones de la Operación de Estudios de Posgrado en el Tecnológico Nacional de México.

León, Guanajuato a 17 del mes de enero de 2022



Manuel Alejandro Torres Fonseca

ACUERDO PARA USO DE OBRA (TESIS DE GRADO)

A QUIEN CORRESPONDA

PRESENTE

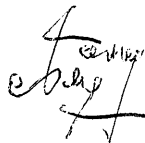
Por medio del presente escrito, **Manuel Alejandro Torres Fonseca** (en lo sucesivo el AUTOR) hace constar que es titular intelectual de la obra denominada: **“Diseño evolutivo de descriptores para la caracterización y reconocimiento de imágenes de expresiones faciales”**, (en lo sucesivo la OBRA) en virtud de lo cual autoriza al Tecnológico Nacional de México/Instituto Tecnológico de León (en lo sucesivo TECN/IT León) para que efectúe resguardo físico y/o electrónico mediante copia digital o impresa para asegurar su disponibilidad, divulgación, comunicación pública, distribución, transmisión, reproducción, así como digitalización de la misma con fines académicos y sin fines de lucro como parte del Repositorio Institucional del TECN/ITLeón.

De igual manera, es deseo del AUTOR establecer que esta autorización es voluntaria y gratuita, y que de acuerdo a lo señalado en la Ley Federal del Derecho de Autor y la Ley de Propiedad Industrial el TECN/IT León cuenta con mi autorización para la utilización de la información antes señalada, estableciendo que se utilizará única y exclusivamente para los fines antes señalados. El AUTOR autoriza al TECN/IT León a utilizar la obra en los términos y condiciones aquí expresados, sin que ello implique se le conceda licencia o autorización alguna o algún tipo de derecho distinto al mencionada respecto a la “propiedad intelectual” de la misma OBRA; incluyendo todo tipo de derechos patrimoniales sobre obras y creaciones protegidas por derechos de autor y demás formas de propiedad intelectual reconocida o que lleguen a reconocer las leyes correspondientes. Al reutilizar, reproducir, transmitir y/o distribuir la OBRA se deberá reconocer y dar créditos de autoría de la obra intelectual en los términos especificados por el propio autor, y el no hacerlo implica el término de uso de esta licencia para los fines estipulados. Nada de esta licencia menoscaba o restringe los derechos patrimoniales y morales del AUTOR.

De la misma manera, se hace manifiesto que el contenido académico, literario, la edición y en general de cualquier parte de la OBRA son responsabilidad de AUTOR, por lo que se deslinda al (TECN/ITLeón) por cualquier violación a los derechos de autor y/o propiedad intelectual, así como cualquier responsabilidad relacionada con la misma frente a terceros. Finalmente, el AUTOR manifiesta que estará depositando la versión final de su documento de Tesis, OBRA, y cuenta con los derechos morales y patrimoniales correspondientes para otorgar la presente autorización de uso.

En la ciudad de León, del estado de Guanajuato a los 17 días del mes de enero de 2022.

Atentamente,



Manuel Alejandro Torres Fonseca

Dedicatoria

*A mis padres, José Manuel Torres Aguirre y Alejandra Fonseca Quiroz, quienes me han inculcado el deseo de adquirir nuevos conocimientos e ir superando me cada día tanto intelectualmente como espiritualmente. Para mí son un ejemplo a seguir y quiero dedicarles esta tesis como una recompensa de todos los esfuerzos que han hecho mí.
Muchas gracias por todo su apoyo los amo.*

Agradecimientos

Al Tecnológico Nacional de México/Instituto Tecnológico de León que me permitió conocer a mis asesores, el Dr. Manuel Ornelas Rodríguez y el Dr. Valentín Calzada Ledesma, a quienes les agradezco mucho por ayudarme a lo largo de mi preparación y por motivarme en cada tropiezo. De igual manera, agradezco a mi comité tutorial, Dr. Alfonso Rojas Domínguez y el Dr. Juan Martín Carpio Valadez por brindarme tanto sus recomendaciones como por sus consejos. Finalmente, al Consejo Nacional de Ciencia y Tecnología (CONACYT) por la beca que se me fue otorgada.

Índice

1. Introducción	1
1.1. Definición del Problema	5
1.2. Hipótesis	6
1.3. Justificación	6
1.4. Objetivo General	7
1.4.1. Objetivos Específicos	7
1.5. Alcances	8
1.6. Organización de la Tesis	9
2. Estado del Arte	11
3. Marco Teórico	18
4. Metodología	33
4.1. Etapa de generación de descriptores	33
4.2. Etapa de evaluación de descriptores	40
5. Experimentación y resultados	41
5.1. Bases de datos	41
5.2. Comparativa entre clasificadores	42
5.3. Esquemas de experimentación	43
5.3.1. Primera experimentación	43
5.3.2. Segunda experimentación	46
5.3.3. Análisis de ocurrencia de operadores en GE	55

6. Conclusiones	59
7. Anexo A	67

Índice de figuras

1.	Sistema de reconocimiento de expresiones faciales [18]	2
2.	Diagrama general para el proceso de mapeo [32]	20
3.	Ejemplo de Gramática BNF.	21
4.	Proceso de mapeo DF para obtener operaciones de números complejos. . .	22
5.	Proceso de mapeo BF [32].	23
6.	Hiperplano de SVM [39].	31
7.	Diagrama de generación de descriptores [5].	34
8.	Diagrama de extracción de características [5].	38
9.	Ejemplo del proceso de caracterización de una imagen.	40
10.	Gráfica de desempeño del descriptor HOG con DE.	49
11.	Gráfica de desempeño del descriptor HOG con SEED.	50
12.	Ejemplos de imágenes de las bases de datos JAFFE, CK+ y KDEF.	53
13.	Gráfica de frecuencias de descriptores con la base de datos JAFFE.	55
14.	Gráfica de frecuencias de descriptores con la base de datos CK+.	56
15.	Gráfica de frecuencias de descriptores con la base de datos KDEF.	57
16.	Gráfica de frecuencias de descriptores con las 3 bases de datos.	58

Lista de algoritmos

1.	Pseudocódigo de Evolución Diferencial	24
2.	Pseudocódigo de SEED	27
3.	Pseudocódigo del método de selección	28

Índice de tablas

1.	Promedio de porcentajes de clasificación [6].	13
2.	Operadores básicos utilizados en la Gramática BNF [5].	36
3.	Reglas de producción utilizadas en la Gramática BNF [5].	37
4.	Resultados de JAFFE y HOG.	43
5.	Comparación del desempeño de diferentes metodologías usando la base de datos JAFFE.	44
6.	Mejor fitness de JAFFE y CK+ con HOG y distintos motores de búsqueda.	46
7.	Parámetros del segundo esquema de experimentación.	47
8.	Porcentaje de clasificación promedio en la etapa de generación de descriptores con el segundo esquema de experimentación.	48
9.	Comparación de resultados de generación y evaluación de descriptores en la base de datos CK+.	51
10.	Comparación de resultados de generación y evaluación de descriptores en la base de datos JAFFE.	51
11.	Comparación de resultados de generación y evaluación de descriptores en la base de datos KDEF.	52
12.	Resultados de clasificación sin utilizar Evolución Gramatical.	54

Acrónimos

ANN	Artificial Neural Network
BEDA	Boltzmann-Estimation Distribution Algorithm
BF	Breadth-First
BG-UNDA	Boltzmann-Gaussian Univariate Marginal Distribution Algorithm
BNF	Backus-Naur Form
BUMDA	Boltzmann-Univariate Marginal Distribution Algorithm
CK+	Extended Cohn-Kanade
COMIA	Congreso Mexicano de Inteligencia Artificial
DE	Differential Evolution
DF	Depth-First
DT	Decision Tree
EDA	Estimation Distribution Algorithm
EMNA-B	Estimation of Multivariate Normal Algorithm con selección de Boltzmann
FER	Facial Expression Recognition
FERET	Face Recognition Technology
GE	Grammatical Evolution
GFC	Gabor Fisher Classifier
GOA	Grasshopper Optimisation Algorithm
HOG	Histogram of Oriented Gradients
ISIC	International Skin Imaging Collaboration
JAFFE	Japanese Female Facial Expression
KDEF	Karolinska Directed Emotional Faces

KNN	k-Nearest Neighbors
LBP	Local Binary Pattern
LGBPHS	Local Gabor Binary Pattern Histogram Sequence
MUG	Multimedia Understanding Group
PCA	Principal Component Analysis
PSO	Particle Swarm Optimization
RAF	Real-world Affective Faces
SEED	Symmetric-Approximation Energy-Based Estimation of Distribution
SVM	Support Vector Machine
WLD	Weber Local Descriptor

Resumen

El reconocimiento de expresiones faciales representa una importante tarea en las áreas de visión por computadora y reconocimiento de patrones. La obtención de características suficientemente discriminativas de imágenes de rostros para reconocer diferentes emociones resulta ser una etapa de suma dificultad. Dentro del estado del arte se encuentran diferentes enfoques que nos permiten analizar dichas imágenes, uno de ellos consiste en procesar únicamente áreas importantes (forma local) como por ejemplo los ojos, cejas, labios, etc. El otro enfoque consiste en el análisis de la imagen completa (forma global). En este trabajo se presenta una metodología de enfoque global basada en Evolución Gramatical y distintos descriptores (HOG, LBP e Histograma de la imagen) para la generación automática de descriptores capaces de caracterizar adecuadamente imágenes de expresiones faciales. Esta metodología se aplicó a las bases de datos de imágenes de expresiones faciales JAFFE, CK+ y KDEF.

Abstract

Recognition of facial expressions represents an important task in areas of computer vision and pattern recognition. Obtaining sufficiently discriminated characteristics to recognize different emotions turns out to be a very difficult stage. In the state of art you can find different perspectives that permit us analyse this images, the first consist in process only important areas (local form) for example the eyes, eyebrows, mouth, etc. the second perspective consist in analyse the complete image (global form). This work presents an global perspective methodology based on Grammatical Evolution and other descriptors (HOG, LBP and Histogram of the image) for the automatic generation of descriptors capable of characterize properly images of facial expressions. This methodology applied at database of images facial expressions JAFFE, CK+ and KDEF.

1. Introducción

Dentro de la comunicación humana, las expresiones faciales toman un rol muy importante al momento de entablar una conversación; un gran porcentaje del mensaje que se quiere comunicar se muestra a través de estas expresiones, por lo que se da un indicio del contexto de la información que se está transmitiendo, así como también del estado de ánimo de la persona.

Actualmente, el estudio formal de las emociones se ha incrementado y existe una gran cantidad de autores que han contribuido en esta área de estudio. Uno de los aportes más importantes fue realizado por Ekman, en su libro titulado: *“The nature of emotion: Fundamental questions”*, en donde se propuso una estandarización de las emociones, formando un grupo de seis emociones básicas: enojo, miedo, disgusto, felicidad, tristeza y sorpresa [8]. La estandarización de las emociones humanas fue uno de los puntos de partida que hicieron posible el reconocimiento de expresiones faciales a nivel computacional, esto permitió un nuevo enfoque para la interacción humano máquina. De esta manera, ha sido posible adaptar herramientas de visión por computadora para el análisis de las emociones, lo que ha permitido el desarrollo de diversos instrumentos que ayudan, por ejemplo, a la identificación de un posible agresor, la predicción del comportamiento de personas en hospitales o instituciones psiquiátricas, sistemas de reconocimiento biométrico, sistemas de seguridad en aeropuertos, etc [6].

Cualquier sistema de reconocimiento de expresiones faciales usualmente se basa en las siguientes fases (ver Figura 1): Adquisición de imágenes de rostros, Extracción de

características y Clasificación.

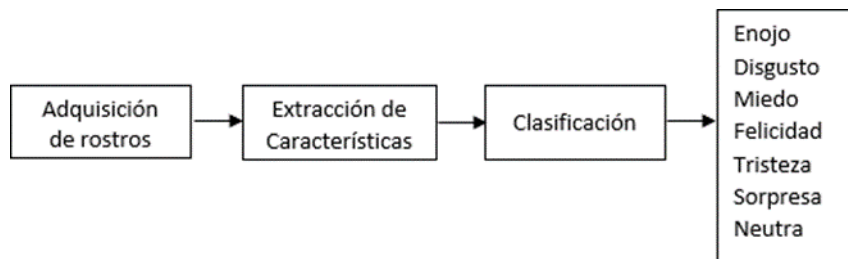


Figura 1: Sistema de reconocimiento de expresiones faciales [18]

La primera fase consiste en la obtención y preprocesamiento de las imágenes de los rostros o secuencias de videos, en el preprocesamiento se realiza un mejoramiento de las imágenes adquiridas previo a la extracción de características, por ejemplo, se puede segmentar y eliminar el fondo de las imágenes para únicamente tomar en cuenta zonas de interés en la imagen. Otro ejemplo, es la aplicación de filtros que permitan eliminar el ruido causado por la iluminación.

La extracción de características es una de las fases más importantes dentro del sistema, usualmente se lleva a cabo a través de la aplicación de descriptores, los cuales son algoritmos computacionales que extraen y empaquetan los atributos significativos de las imágenes en vectores de características. Es deseable que estos atributos caracterizados permitan una buena separabilidad entre clases (emociones), permitiendo al clasificador tener un buen desempeño. Finalmente, en la fase de clasificación, los vectores de características se etiquetan o asocian a una clase cuyas características son similares; esto se realiza mediante los algoritmos clasificadores, los cuales utilizan un indicador (porcentaje de clasificación) que permite conocer la cantidad de vectores que fueron asociados con sus

respectivas clases de manera correcta.

Existe una variedad de descriptores que pueden ser utilizados para caracterizar imágenes, por ejemplo, Patrones Binarios Locales (LBP por sus siglas en inglés), Filtros de Gabor, Histograma de Gradientes Orientados (HOG por sus siglas en inglés), entre otros. Estos algoritmos han sido diseñados por expertos en el campo para abordar un determinado tipo de problemas, sin embargo, es común observar que el rendimiento de éstos puede variar dependiendo del tipo de problema. Lo mencionado anteriormente, se observa en los resultados de la investigación realizada por Alhindi et al. [2] en su artículo titulado “*Comparing LBP, HOG and Deep Features for Classification of Histopathology Images*”, en donde diferentes descriptores, tales como: LBP, HOG y una red profunda pre entrenada (VGG19), se utilizaron para caracterizar imágenes histopatológicas (imágenes de tomas microscópicas de tejidos). Los algoritmos encargados de la clasificación fueron: Máquinas de vector soporte (SVM por sus siglas en inglés), arboles de decisión (DT por sus siglas en inglés) y redes neuronales artificiales (ANN por sus siglas en inglés). Los resultados de clasificación empleando validación cruzada (1, 2 y 3 pliegues) en el caso del descriptor LBP, obtuvieron el mejor resultado empleando los clasificadores SVM ($90.52\% \pm 1.26\%$), con el descriptor HOG el mejor resultado fue de ANN ($36.15\% \pm 0.15\%$) y usando VGG19 el mejor resultado fue con el clasificador DT ($47.18\% \pm 1.42\%$). Los resultados obtenidos mediante HOG indicaron que no es recomendable aplicar este descriptor a este tipo de imágenes.

En el artículo de Xiang et. al. [41] titulado: “*The excellent properties of a dense grid-based HOG features on face recognition compared to Gabor and LBP*” se demostró un

desempeño distinto del HOG al aplicarse a imágenes de rostros de la base de datos Facial Recognition Technology (FERET), en el cual se usaron 4 variaciones de la base de datos FERET (fb, fc, dup1 y dup2) y se compararon 3 descriptores diferentes (HOG, LBP y Gabor Fisher Classifier). Con la primera variación del FERET(fb) el mejor porcentaje alcanzado fue por el descriptor Gabor Fisher Classifier (GFC) con un 96.3 % mientras que HOG alcanzó un 92.6 %, con la segunda variación (fc) el mejor porcentaje fue de GFC con un 81.4 % mientras que HOG alcanzó un 72.2 %, con la tercera variación (dup1) GFC fue el mejor con un 68.8 % mientras que HOG alcanzó un 66.3 % y la última variación (dup2) HOG con un 61.1 % superó a GFC (49.1 %). Con lo mostrado anteriormente se puede decir que no existe un descriptor perfecto.

Este hecho ha motivado la creación de metodologías para la generación automática de descriptores que se adapten a un conjunto de imágenes en particular para poder obtener un mejor desempeño en el proceso de clasificación [5]. En la presente investigación se propuso una metodología basada en Evolución Gramatical (GE por sus siglas en inglés), para la generación automática de descriptores aplicados al reconocimiento de expresiones faciales, utilizando como instancia de prueba diferentes bases de datos de imágenes, tales como: Japanese Female Facial Expression(JAFFE), Extended Cohn-Kanade (CK+) y Karolinska Directed Emotional Faces (KDEF). Los algoritmos de reconocimiento de patrones que se encargaron de la clasificación fueron k-Vecinos más Cercanos (K-NN por sus siglas en inglés) y Máquinas de Vector Soporte (SVM).

1.1. Definición del Problema

El reconocimiento de expresiones faciales (comunicación no verbal) y el tono de voz (comunicación verbal), son indicadores que permiten conocer el estado de ánimo de una persona [5]. Al conocer el estado de ánimo de una persona es posible tomar decisiones importantes que influyen en su desempeño al realizar algunas tareas. Dentro de la interacción humano-máquina las expresiones faciales proporcionan la información necesaria para poder detectar el estado de ánimo de una persona. En el reconocimiento de expresiones faciales mediante el uso de sistemas computarizados de visión se consideran las mismas variaciones que en el reconocimiento de rostros. Debido a esto, el sistema debe ser robusto ante alteraciones de las imágenes como cambios de iluminación, desplazamientos, ruido, por mencionar algunas [37]. En los sistemas de reconocimiento de emociones como en cualquier sistema de reconocimiento de imágenes, es difícil conocer cuáles descriptores son los más adecuados para la extracción de características. Este problema es crítico, porque las características extraídas (vectores de características) influyen en el desempeño y en la eficiencia del algoritmo de reconocimiento de patrones encargado de clasificar las imágenes. Por ello, se requiere la intervención de un experto del tema para poder seleccionar los descriptores más adecuados para una aplicación en específico.

Con base en el problema planteado, se empleó el método de Evolución Gramatical para la generación automática de descriptores adaptados al problema de reconocimiento de expresiones faciales, sin la necesidad de la intervención de un experto en el tema. Se usó este método con la finalidad de responder la pregunta: ¿Podrá el método Evolución Gramatical proporcionar descriptores que ayuden a la obtención de vectores de características

que favorezcan la clasificación de las imágenes de expresiones faciales?

1.2. Hipótesis

Es posible generar automáticamente descriptores de imágenes de expresiones faciales empleando Evolución Gramatical, los cuales permitan obtener resultados de clasificación comparables con los reportados en el estado del arte.

1.3. Justificación

El reconocimiento de emociones es un problema importante ya que tiene aplicabilidad en diferentes sectores, como es el caso de la robótica, en la comunicación, análisis psicológico de una persona, etc. [33]. Las emociones proporcionan datos sobre el comportamiento humano, y al poder identificarlo se puede realizar un análisis, el cual permita tomar decisiones que ayuden a prevenir una posible situación de riesgo.

Otra aplicación del reconocimiento de emociones en el ámbito social es el propuesto por V. Franzoni et al. [12] en su artículo titulado: *“Reconocimiento de emociones para la autoayuda en los tratamientos de adicciones, terapias psicológicas y comunicación no violenta”*. En el caso de tratamientos para las adicciones V. Franzoni et al. explican cómo puede ayudar a las instituciones encargadas de la rehabilitación de personas adictas, una herramienta que permita reconocer las emociones de sus pacientes, debido a que las personas con adicciones suelen tener cambios de humor repentinos. Al conocer los cambios de humor mediante el monitoreo de las emociones de los pacientes, los encargados podrán obtener la información acerca del tipo de tratamiento que son aplicables al pa-

ciente basándose en su comportamiento [37].

El enfoque para tratar el tema de reconocimiento de emociones fue mediante el uso del método Evolución Gramatical, pues los resultados proporcionados en el artículo de Calzada [5], han demostrado que esa metodología permite resolver adecuadamente problemas de reconocimiento de texturas, por lo que se aplicó al reconocimiento de emociones en imágenes de expresiones faciales.

1.4. Objetivo General

Desarrollar un sistema computacional basado en Evolución Gramatical para la generación automática de descriptores aplicados al reconocimiento de imágenes de expresiones faciales.

1.4.1. Objetivos Específicos

- Investigar el estado del arte relacionado con el tema de investigación.
- Obtener bases de datos de imágenes del estado del arte, con las cuales se va a trabajar.
- Identificar e implementar la base de descriptores de imágenes a utilizar en el proceso evolutivo.
- Proponer una metodología basada en Evolución Gramatical para el diseño automático de los descriptores.
- Implementar la metodología diseñada y realizar la experimentación.

- Realizar un análisis estadístico de los resultados obtenidos.

1.5. Alcances

- Se diseñó e implementó una metodología basada en Evolución Gramatical para la generación automática de descriptores aplicados al reconocimiento de imágenes de expresiones faciales.
- Se usaron el 100 % de las imágenes de las bases de datos JAFFE y CK+ y de la base de datos KDEF solamente se utilizó el 10 % para probar la metodología diseñada.
- Se empleó la misma base de filtros de procesamiento de imágenes usados en el artículo de Valentín Calzada [5].
- Se programó en java empleando librerías de OpenCV, WEKA, MathCommons, etc.
- Se aplicaron los descriptores creados por la Gramática a las imágenes y se obtuvieron sus características mediante LBP, HOG y el histograma de la imagen.
- Se implementaron los motores de búsqueda Evolución Diferencial y SEED.
- Se emplearon los algoritmos de reconocimiento de patrones K-NN y SVM para clasificar las características obtenidas.
- Se realizaron análisis de los resultados empleando las distintas combinaciones entre los motores de búsqueda (Evolución Diferencial y SEED) con los distintos descriptores (LBP, HOG y el histograma) y las distintas bases de datos (JAFFE, CK+ y KDEF).

1.6. Organización de la Tesis

Capítulo 2 Estado del Arte

En este capítulo se encuentran diversas propuestas consultadas en la literatura de proyectos relacionados, que permiten conocer ampliamente lo que se ha realizado respecto al tema de investigación, para poder realizar una comparación.

Capítulo 3 Marco Teórico

En este capítulo se muestran conceptos y definiciones importantes que están involucrados en la tesis, tal como Algoritmo Evolutivo, Motor de Búsqueda, Algoritmos de reconocimiento de patrones, etc.

Capítulo 4 Metodología

En este capítulo se explica el método propuesto basado en Evolución Gramatical para la caracterización de las imágenes de expresiones faciales.

Capítulo 5 Experimentación y resultados

En este capítulo se explican los parámetros empleados en los motores de búsqueda, descriptores y algoritmos de reconocimiento de patrones. Así también, se muestran los resultados obtenidos a partir de dichos parámetros y una comparativa entre los resultados obtenidos.

Capítulo 6 Conclusiones y trabajo a futuro

En este capítulo se presentan las conclusiones del trabajo realizado, así como algunas propuestas que ayuden a enriquecer la metodología implementada.

Anexo A Artículo publicado en el COMIA

En este anexo se muestra el artículo publicado en el Congreso Mexicano de Inteligencia Artificial (COMIA 2020) titulado: "*Diseño de descriptores mediante Evolución Gramatical para el reconocimiento de imágenes de expresiones faciales*", así también se incluye el certificado por la participación a dicho congreso.

2. Estado del Arte

En este apartado se mencionarán algunas investigaciones que forman parte del estado del arte relacionadas con el tema del reconocimiento de emociones mediante imágenes de expresiones faciales, estas investigaciones servirán para ejemplificar las distintas maneras en que se puede realizar la extracción de las características de una imagen. También se hará referencia a algunos clasificadores conocidos y empleados en el ámbito de reconocimiento de patrones.

El reconocimiento de expresiones faciales nos permite analizar el rostro con diferentes enfoques, uno de ellos es analizando únicamente áreas importantes (forma local) como es el caso de los ojos, cejas, labios, etc. La otra alternativa es analizando la imagen completa (forma global). A continuación se describirán algunas investigaciones que discuten estos análisis.

En el artículo de Santosh et al. [34] se propone una metodología en la que se emplea el descriptor Histograma de Orientación de Gradiente (HOG por sus siglas en inglés) y Análisis de Componentes Principales (PCA por sus siglas en inglés) para la extracción de las características de las imágenes de la base de datos JAFFE. Se obtuvo un porcentaje de clasificación del 90.5% empleando el algoritmo Mínima Distancia con la distancia Canberra.

En el artículo de Adouani et al. [1] realiza una comparación entre los descriptores: características Haar, HOG y LBP aplicados a la base de datos de secuencias de imágenes

DEAP clasificadas por SVM, el rendimiento de los descriptores fueron los siguientes: Haar 78.23 %, HOG 92.68 % y LBP 60.37 %.

En el artículo de Paraneswaran et al. [30] titulado: *“Exploring Various Aspects of Gabor Filter in Classifying Facial Expression”*, emplearon bancos de filtros de Gabor con distintos anchos de banda y orientaciones combinadas con SVM para la clasificación de imágenes de expresiones faciales de la base de datos CK+. De esta se manera se alcanzó un porcentaje de clasificación del 91 %.

En el artículo de Jain et al. [15] emplearon una Red Neuronal Convolutacional Profunda para el reconocimiento de las expresiones faciales de las bases de datos CK+ y JAFFE, la arquitectura de la Red Neuronal Profunda esta formada de la siguiente manera: 6 capas convolucionales, 2 bloques de aprendizaje residual profundo y 2 capas las cuales están completamente conectadas. Las imágenes de las bases de datos antes de entrar a la Red Neuronal fueron preprocesadas (filtro a escala de grises) y segmentadas (recortadas a únicamente el rostro), obtuvieron un rendimiento del 93.24 % para la base de datos CK+ y un 95.23 % para la JAFFE.

En el artículo de Fan et al. [11] elaboraron un sistema de reconocimiento de emociones basado en video, este sistema se creó mediante una combinación de redes neuronales recurrentes y redes convolucionales 3D, las redes neuronales recurrentes obtuvieron las apariencias de las características de cada frame de entrada, posteriormente las redes convolucionales 3D se encargaron de modelar la apariencia y la emoción del video, obteniendo una precisión del 59.02 %.

En el artículo de Cornejo et al. [6] titulado: “*Emotion recognition from occluded facial expressions using Weber local descriptor*”, se implementó una metodología aplicada al reconocimiento de rostros tomando en cuenta imágenes de las bases de datos CK+, JAFFE Y MUG. Los descriptores que implementaron en combinación con el Análisis de Componente Principales (PCA por sus siglas en inglés) y Análisis de Discriminante Lineal (LDA por sus siglas en inglés) fueron: Descriptor Local de Weber (WLD por sus siglas en inglés), Patrones Locales Binarios (LBP por sus siglas en inglés) y HOG. En la tabla 1 se muestran los resultados obtenidos tomando en cuenta los clasificadores SVM y KNN, en donde los valores indicados en negritas representan el mejor desempeño entre WLD, LBP y HOG en cada una de las metodologías implementadas.

Tabla 1: Promedio de porcentajes de clasificación [6].

	Recognition Method	Representation		
		WLD	LBP	HOG
CK+	PCA + K-NN	51.00	43.74	64.63
	PCA + LDA + K-NN	93.13	92.62	90.45
	PCA + SVM	85.45	77.17	83.36
	PCA + LDA + SVM	93.44	92.84	91.20
JAFFE	PCA + K-NN	87.44	64.41	83.10
	PCA + LDA + K-NN	95.35	93.00	96.43
	PCA + SVM	90.70	84.18	92.74
	PCA + LDA + SVM	95.35	92.50	95.60
MUG	PCA + K-NN	86.00	79.69	80.89
	PCA + LDA + K-NN	93.09	91.40	91.02
	PCA + SVM	90.12	85.70	85.26
	PCA + LDA + SVM	93.09	90.07	89.12

Las investigaciones reportadas anteriormente implementaron herramientas que permiten la extracción de las características de la imagen sin la necesidad de analizar áreas específicas del rostro, por el contrario, existen otras herramientas de extracción de características (características Haar, Modelo de Contorno Activo, etc.), las cuales buscan zonas de interés dentro de la imagen. Estas herramientas, conocidas como descriptores locales son aplicados al reconocimiento de expresiones faciales. Este tipo de descriptores se implementaron en las siguientes investigaciones.

En el artículo de Juliana et al. [17] a partir de una secuencia de video se fue tomando cada imagen para hacer el preprocesamiento, la extracción de las características y la clasificación. El preprocesamiento consistió en dividir los ojos, la boca y la nariz de cada imagen, posteriormente se extrajeron sus características empleando HOG o LBP y finalmente las características fueron clasificadas mediante una Red Neuronal. De esta manera se obtuvo un rendimiento del 87 % (para HOG) y 64 % (para LBP) en promedio.

En el artículo de Turan et al. [38] titulado: "*Histogram-based local descriptors for facial expression recognition (FER): A comprehensive study*", se hace una comparativa entre distintos descriptores locales como es el caso de: Weber Local Descriptor (WLD), Local Arc Pattern (LAP), Local Binary Pattern (LBP), Local Directional Pattern (LDP), etc. Estos descriptores fueron aplicados a las siguientes bases de datos de imágenes: JAFFE, CK+, BAUNM-2 y TFEID. Mediante el clasificador K-NN, los mejores resultados por base de datos fueron los siguientes: JAFFE 92.24 % con el descriptor WLD, CK+ 95.90 % con el descriptor WLD, BAUNM-2 62.41 % con Local Gabor Binary Pattern Histogram Sequence (LGBPHS), TFEID 95.15 % con LGBPHS.

En el artículo de Mohan et al. [28] titulado: “*Facial Expression Recognition Using Local Gravitational Force Descriptor-Based Deep Convolution Neural Networks*”, se usa el descriptor Local Gravitational Force en combinación con una Red neuronal convolucional profunda para la clasificación de las imágenes de expresiones faciales de las siguientes bases de datos: FER2013, JAFFE, CK+, KDEF y RAF. Los resultados obtenidos fueron los siguientes: FER2013 76.27 %, JAFFE 94.38 %, CK+ 96.79 %, KDEF 94.12 % y RAF 79.27 %.

En el artículo de Lakshmi et al. [19] titulado: “*Facial emotion recognition using modified HOG and LBP features with deep stacked autoencoders*”, se realizó la detección de áreas importantes del rostro (ojos, nariz y boca) empleando el algoritmo de Viola Jones y el Filtro pasa alta de Butterworth, las áreas segmentadas se caracterizaron empleando el HOG mejorado y LBP. Las características obtenidas se redujeron empleando el algoritmo autocodificadores apilados profundos para que finalmente fueran clasificadas empleando SVM multiclase. Esta metodología fue aplicada a las bases de datos JAFFE y CK+ obteniendo un rendimiento del 97.67 % (JAFFE) y 97.34 % (CK+) respectivamente.

En el artículo de Yang et al. [42] se segmentaron y se recortaron las imágenes de la base de datos JAFFE para tomar en cuenta únicamente la zona de los ojos y de la boca empleando las cascadas Haar, posteriormente se les aplicó el filtro a escala de grises y el método de detección de contornos de Sobel, a partir del método de Sobel se obtuvieron los vectores de características que alimentaron la Red Neuronal, los porcentajes de clasificación por emoción fueron los siguientes: tristeza 80.5 %, disgusto 89.2 %, miedo 84.3 %,

enojo 84.3% y felicidad 90.0%.

Las investigaciones mencionadas anteriormente emplean herramientas o algoritmos, los cuales requieren parámetros para su correcto funcionamiento. Al momento de crear un sistema, es común preguntarse cuáles serán los mejores valores que requiere este algoritmo para que proporcione resultados óptimos. Este problema puede ser abordado mediante el uso de algoritmos de optimización inteligente, los cuales calibran los parámetros de alguna herramienta mediante procesos iterativos. Dentro de los algoritmos de optimización inteligente se encuentran los algoritmos evolutivos y bioinspirados. A continuación, se muestran algunas de estas investigaciones.

En el artículo de Boubenna et al. [4] se desarrolló un sistema de reconocimiento de expresiones faciales aplicando algoritmos genéticos en combinación con la herramienta análisis de discriminación lineal para la extracción de las características de las imágenes, el descriptor empleado fue HOG piramidal el cual requiere la cantidad de capas que se emplearán para la obtención de las celdas que indicarán el área en que será tomado el gradiente y así obtener el histograma del gradiente, esto implica que la cantidad de capas usadas indica el tamaño de los vectores de características, por esta razón se emplearon los algoritmos genéticos para calibrar la cantidad de capas empleadas por HOG piramidal. Los vectores de características fueron clasificados con una red neuronal obteniendo una precisión del 98.67%.

En el artículo de Malika et al. [3] titulado: "*Facial Emotion Recognition System Based on PCA and Gradient Features*" se desarrolló un sistema de reconocimiento de expre-

siones faciales aplicados a la base de datos JAFFE. Se empleó un sistema híbrido para la extracción de las características en la cual intervinieron el filtro de gradiente, Análisis de Componentes Principales (PCA) y Particle Swarm Optimization (PSO). Este sistema aplicó el filtro de gradiente a las imágenes y PSO se encargó de optimizar la cantidad de características que serán obtenidas a partir de PCA para finalmente clasificarlas mediante Random Forest, el resultado promedio de la base de datos JAFFE fue del 91.3 %.

En el artículo de Ibrahim et al. [14] se empleó el algoritmo de optimización del saltamontes (GOA por sus siglas en inglés) para optimizar los parámetros de SVM, GOA se encargó de optimizar los parámetros C y γ del clasificador SVM que fue aplicado a la base de datos de pacientes con cáncer iraquíes del 2010 al 2012. Los resultados de clasificación de SVM con y sin GOA fueron los siguientes: cáncer (2010) sin GOA 76.44 %, cáncer (2010) con GOA 97.60 %, cáncer (2011) sin GOA 90.68 %, cáncer (2011) con GOA 99.9 %, cáncer (2012) sin GOA 89.25 % y cáncer (2012) con GOA 100 %.

En el artículo de Tan et al. [36] titulado: “*Evolving Ensemble Models for Image Segmentation Using Enhanced Particle Swarm Optimization*” se propuso una segmentación de imágenes de manchas en la piel empleando una red neuronal profunda y un modelo de clustesrizado (Fuzzy C-Means). La red neuronal fue optimizada empleando Particle Swarm Optimization (PSO) y DE, y se aplicó a las bases de datos PH2, ISIC 2017 y Dermofit Image Library.

3. Marco Teórico

En este capítulo se aborda el tema de Evolución Gramatical y cada uno de los elementos que lo conforman, los cuales son: Gramática libre de contexto, proceso de mapeo, el motor de búsqueda, el descriptor y el clasificador.

Evolución Gramatical genera soluciones mediante los principios de la Evolución Darwiniana la cual dice que los individuos que mejor se adaptan a su ambiente son los que van sobreviviendo[10]. Dentro de cómputo evolutivo, los algoritmos genéticos y la programación genética son sus 2 formas más conocidas. Los algoritmos genéticos emplean poblaciones de individuos en donde cada uno de ellos es una cadena de bits. Para encontrar una solución apta al problema que se planteó se realiza un proceso evolutivo. La programación genética emplea poblaciones de individuos que son manipulados como árboles de derivación, los cuales emplean una gramática la cual permite su evolución en el dominio del problema [10].

La programación genética requiere un mapeo entre la población de individuos y la gramática la cual permite la obtención de una solución al problema, el mapeo realizado por la programación genética se realiza de uno a uno entre los árboles de derivación.

Evolución Gramatical (GE): es considerada una variación de la Programación Genética [29], la cual utiliza una gramática formal relacionada con el problema, la gramática describe la composición y la estructura de las variables, operadores y restricciones que interfieren en el proceso de creación de programas también denominados fenotipos capa-

ces de obtener la solución al problema que se está aplicando [29]. Para la obtención del fenotipo se requiere una cadena simple (genotipo) de valores binarios o enteros que se irá relacionando con la gramática mediante un esquema de mapeo, este esquema de mapeo efectúa un análisis de cada elemento del genotipo (codón) y algunas reglas para de esta manera formar el fenotipo [29].

Dentro de GE es necesario tomar en cuenta las reglas o restricciones que requieren los fenotipos, las cuales están implícitas en la estructura de la gramática, la recursividad toma un rol muy importante en la estructuración de la gramática permitiendo agregar las reglas y restricciones para la creación del fenotipo que se desea obtener. La flexibilidad que permite la gramática de GE para adaptarse a distintos tipos de problemas es una de sus principales ventajas, pues se puede adaptar a distintos ambitos que pueden ser desde la creación de descriptores capaces de caracterizar imágenes hasta la aplicación en la música (música evolutiva) [20]. A continuación, se presentarán cada uno de los elementos que conforman GE.

Gramática libre de contexto tipo Backus-Naur (BNF): es una notación formal que permite diseñar la sintaxis de un programa basándose en un conjunto de reglas de producción [29], estas reglas permiten la obtención del programa (fenotipo) tomando en cuenta símbolos divididos en terminales y no terminales. Los elementos que conforman la gramática son expresados mediante la tupla N, T, P, S , donde N representa los símbolos no terminales (indicados en la gramática entre signos ' $<$ ' y ' $>$ '), T indica los símbolos terminales, P es el conjunto de reglas de producción y S es el símbolo inicial que indica la primera regla de producción que se aplica [32].

Proceso de Mapeo: se encarga de convertir un cromosoma genotípico en un fenotipo basándose en los elementos que conforman la gramática BNF (N, T, P, S) ; cada elemento del cromosoma recibe el nombre de codón. A continuación, se mencionarán algunos procesos de mapeo revisados en el estado del arte los cuales se basan en el esquema general de mapeo de la Fig. 2.

El primer algoritmo es el Depth-First (DF), el cual utiliza una regla de derivación que recorre los símbolos no terminales de la gramática hasta encontrar símbolos terminales. En la Figura 2 se puede observar la regla de derivación y el proceso de mapeo, el cual recibe un codón para aplicar la regla a la gramática e ir formando el árbol de derivación hasta obtener el fenotipo.

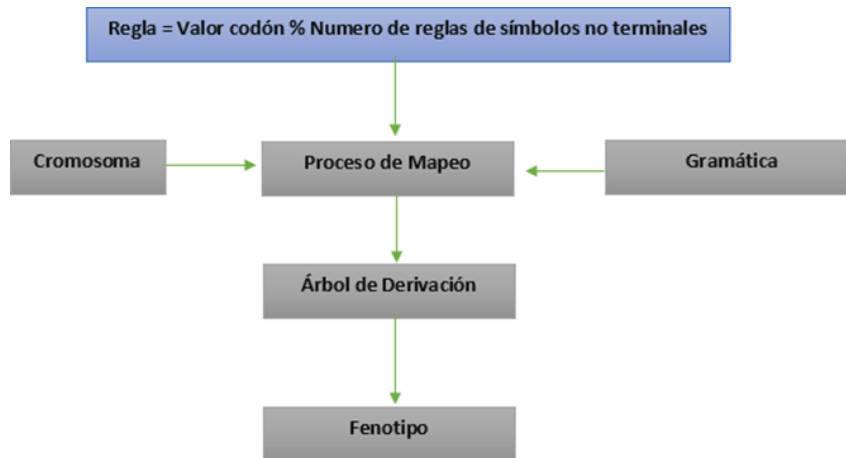


Figura 2: Diagrama general para el proceso de mapeo [32]

A continuación, se muestra un ejemplo de este proceso de mapeo, utilizando la gramática BNF de la Figura 3, la cual se encarga de la creación de operaciones entre números

complejos. En la gramática el símbolo inicial S es: $\langle NumComplejo \rangle$, los terminales T son: $i, +, -, *, /, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0$, los símbolos no terminales N son: $\langle oper \rangle, \langle op \rangle, \langle signo \rangle, \langle OpReal \rangle, \langle numero \rangle$ y el conjunto de reglas de producción P se muestra en la Figura 3.

```

<NumComplejo> ::= (<oper>)<op>(<oper>)    (0)
<oper> ::= <signo> <OpReal><signo> <OpReal> i    (0)
<signo> ::= + |    (0)
           - |    (1)
<OpReal> ::= <OpReal><numero> |    (0)
            <numero> |    (1)
<numero> ::= 1 |    (0)
            2 |    (1)
            3 |    (2)
            4 |    (3)
            5 |    (4)
            6 |    (5)
            7 |    (6)
            8 |    (7)
            9 |    (8)
            0 |    (9)
<op> ::= + |    (0)
        - |    (1)
        * |    (2)
        / |    (3)

```

Figura 3: Ejemplo de Gramática BNF.

El proceso de mapeo DF comienza con el símbolo inicial y procede a sustituir el símbolo no terminal localizado más a la izquierda, usando la regla de derivación basada en el operador módulo, hasta que encuentra un símbolo terminal [32]. Enseguida se desplaza hacia la derecha para sustituir el siguiente símbolo no terminal. Este proceso continúa hasta que todos los símbolos no terminales del fenotipo generado son reemplazados por símbolos terminales

En el ejemplo mostrado en la Figura 4, el primer codón del cromosoma es 15 y el símbolo no terminal inicial tiene solo una regla de producción; aplicando la regla de derivación se obtiene $15 \%1=0$, por lo que se selecciona la producción 0: $(\langle oper \rangle) \langle op \rangle (\langle oper \rangle)$. El siguiente codón es 3 y el símbolo no terminal más a la izquierda $(\langle oper \rangle)$ tiene también una producción válida, dando como resultado $3 \%1=0$, seleccionando $\langle signo \rangle \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i$. El siguiente codón es 31 y el símbolo no terminal más a la izquierda $(\langle signo \rangle)$ tiene dos producciones válidas, dando $31 \%2=1$ y seleccionando “-”. Como se mencionó anteriormente, el proceso de mapeo finaliza cuando ya no existen símbolos no terminales en el fenotipo.

Cromosoma	15	3	31	25	61	48	7	5	12	16	3	17	36	22	41	18	
$(\langle oper \rangle) \langle op \rangle (\langle oper \rangle)$																	15%1=0
$(\langle signo \rangle \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i) \langle op \rangle (\langle oper \rangle)$																	3%1=0
$(- \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i) \langle op \rangle (\langle oper \rangle)$																	31%2=1
$(- \langle numero \rangle \langle signo \rangle \langle OpReal \rangle i) \langle op \rangle (\langle oper \rangle)$																	25%2=1
$(- 2 \langle signo \rangle \langle OpReal \rangle i) \langle op \rangle (\langle oper \rangle)$																	61%10=1
$(- 2 + \langle OpReal \rangle i) \langle op \rangle (\langle oper \rangle)$																	48%2=0
$(- 2 + \langle numero \rangle i) \langle op \rangle (\langle oper \rangle)$																	7%2=1
$(- 2 + 6 i) \langle op \rangle (\langle oper \rangle)$																	5%10=5
$(- 2 + 6 i) + (\langle oper \rangle)$																	12%4=0
$(- 2 + 6 i) + (\langle signo \rangle \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i)$																	16%1=0
$(- 2 + 6 i) + (- \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i)$																	3%2=1
$(- 2 + 6 i) + (- \langle numero \rangle \langle signo \rangle \langle OpReal \rangle i)$																	17%2=1
$(- 2 + 6 i) + (- 7 \langle signo \rangle \langle OpReal \rangle i)$																	36%10=6
$(- 2 + 6 i) + (- 7 + \langle OpReal \rangle i)$																	22%2=0
$(- 2 + 6 i) + (- 7 + \langle numero \rangle i)$																	41%2=1
$(- 2 + 6 i) + (- 7 + 9 i)$																	18%10=8

Figura 4: Proceso de mapeo DF para obtener operaciones de números complejos.

El segundo proceso de mapeo es el Breadth-First (BF), el cual va expandiendo el árbol de derivación tomando primero en cuenta los elementos que se encuentran más a la izquierda de cada nivel (ver figura 5).

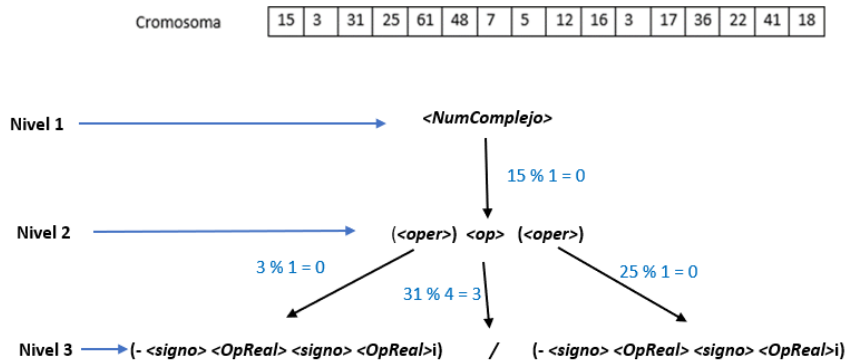


Figura 5: Proceso de mapeo BF [32].

En la figura 5 se puede observar que en el nivel 2 tenemos 3 símbolos no terminales, al ser tomados los símbolos de izquierda a derecha al símbolo ($\langle \text{oper} \rangle$) le corresponde el codón con valor 3, obteniendo como resultado los elementos que se encuentran en la izquierda del nivel 3 ($- \langle \text{signo} \rangle \langle \text{OpReal} \rangle \langle \text{signo} \rangle \langle \text{OpReal} \rangle i$), al segundo símbolo no terminal del nivel 2 ($\langle \text{op} \rangle$) le corresponde el codón con valor de 31, obteniendo como resultado el elemento $/$ que se encuentra en el tercer nivel. Se sigue haciendo el mismo procedimiento hasta derivar todos los símbolos no terminales del nivel 2 y posteriormente se pasa al siguiente nivel, este mapeo concluye cuando tenemos únicamente nodos terminales en el último nivel o cuando ya se recorrió por completo el cromosoma.

Motor de Búsqueda: consiste en un algoritmo de optimización capaz de encontrar soluciones adecuadas dentro de un determinado espacio de búsqueda. Uno de los algoritmos empleados fue Evolución Diferencial (DE por sus siglas en inglés). Para obtener una solución óptima, DE evoluciona un conjunto de soluciones candidatas con el fin de encontrar la solución mejor adaptada al problema. A cada solución se le conoce como individuo y al conjunto de individuos se le denomina población [35]. A continuación, en

el algoritmo 1 se muestra el pseudocódigo de Evolución Diferencial.

Algoritmo 1 Pseudocódigo de Evolución Diferencial

Entrada: F = coeficiente de mutación $\in [0,4,0,9]$

C = coeficiente de cruza $\in [0,1,1]$

Salida: población de individuos x_i

```
1: mientras el criterio de paro sea falso hacer
2:   para cada individuo  $x_i$  desde  $i=1$  hasta  $N$  hacer
3:      $r1 \leftarrow$  entero aleatorio  $\in [1, N] : r1 \neq i$ 
4:      $r2 \leftarrow$  entero aleatorio  $\in [1, N] : r2 \neq \{i, r1\}$ 
5:      $r3 \leftarrow$  entero aleatorio  $\in [1, N] : r3 \neq \{i, r1, r2\}$ 
6:      $v_i \leftarrow x_{r1} + F(x_{r2} - x_{r3})$  vector mutado
7:      $J_r \leftarrow$  entero aleatorio  $\in [1, N]$ 
8:     para cada dimensión desde  $j=1$  hasta  $N$  hacer
9:        $r_{cj} \leftarrow$  numero aleatorio  $\in [0, 1]$ 
10:      si  $r_{cj} < C$  o  $j = J$  entonces
11:         $v_{ij} \leftarrow v_{ij}$ 
12:      si no
13:         $v_{ij} \leftarrow x_{ij}$ 
14:      fin si
15:    fin para
16:  fin para
17:  para cada población desde  $i = 1$  hasta  $N$  hacer
18:    si  $f(v_i) < f(x_i)$  entonces
19:       $x_i \leftarrow v_i$ 
20:    fin si
21:  fin para
22: fin mientras
23: devolver población  $x_i$ 
```

Evolución Diferencial encuentra la solución óptima a través de un método iterativo, el cual a partir de una población de soluciones candidatas, crea otra población. La segunda población se crea aplicándole a la primera población métodos de mutación y cruza, las dos poblaciones son evaluadas a partir de una función de aptitud (la función nos proporciona un valor, el cual nos indica cuál individuo se acerca más a la solución que buscamos) que nos permitirá elegir los individuos mejor adaptados al problema (maximización o minimización) que conformarán la nueva población para la siguiente iteración.

El Algoritmo 1 muestra la forma clásica del algoritmo Evolución Diferencial, donde la mutación (que consiste en la creación de nuevos individuos a partir de un grupo de individuos) se realiza seleccionando aleatoriamente tres individuos (x_{r1}, x_{r2}, x_{r3}) y apli-

cando la operación vectorial $x_{r1} + F(x_{r2} - x_{r3})$, en donde F es el coeficiente de Muta. La Cruza permite la combinación de los elementos de dos individuos para la creación de un individuo distinto, esto se realiza a partir de un coeficiente de cruza el cual es un número entre 0 y 1 (por lo general se usan cantidades altas como 0.9), por cada elemento de los individuos se genera un número aleatorio entre 0 y 1 que se compara con el coeficiente de cruza, si el valor aleatorio es mayor al coeficiente de cruza se toma el elemento del primer individuo y se pasa al nuevo individuo, de otra forma el elemento del segundo individuo es el que se pasa al nuevo individuo.

Otro tipo de algoritmos evolutivos son los algoritmos de estimación de distribución (EDAs), los EDAs son metaheurísticas las cuales construyen modelos probabilísticos mediante un proceso iterativo el cual de manera incremental va obteniendo mejores resultados aplicado a un problema en específico [7]. Los EDAs permiten abarcar problemas de optimización en donde las variables son independientes unas de otras permitiendo crear poblaciones de soluciones las cuales se van comparando y seleccionando para crear un modelo probabilístico a partir de las mejores soluciones obtenidas [7].

Dentro de los EDAs existen diferentes tipos, algunos de ellos son: Boltzmann-EDA (BEDA), Boltzmann-Gaussian Univariate Marginal Distribution Algorithm (BG-UNDA), Estimation of Multivariate Normal Algorithm con selección de Boltzmann (EMNA-B), Boltzmann-UMDA (BUMDA) y Symmetric-Approximation Energy-Based Estimation of Distribution (SEED) [7]. El EDA que se implementó y se acopló a Evolución Gramatical fue SEED, a continuación, se muestran las ecuaciones necesarias para la implementación de SEED.

$$\mu \approx \frac{\frac{1}{Z\beta} \sum_i^m \exp(\beta g_{x_i}) x_i + \sum_i^m g_{x_i} x_i}{\frac{m}{(\beta(b-a))} + \sum_i^m g_{x_i}} \quad (1)$$

En la Ecuación (1) [7] se observa la fórmula que aproxima el valor de la media en donde m representa la cantidad de elementos dentro de una población de soluciones, g_{x_i} es el resultado de evaluar una solución dentro del espacio, a y b corresponden al dominio empleado por el método Monte Carlo para la minimización de la divergencia, Z es un valor que requiere el método Monte Carlo para su correcto funcionamiento el cual se puede obtener mediante la Ecuación (2) [7].

$$Z = 1 / \int \beta g_x dx \approx \frac{m}{((b-a) \sum_i^m \exp(\beta g_{x_i}))} \quad (2)$$

La desviación estándar se obtiene a partir de la Ecuación (3) en donde los parámetros mencionados anteriormente se repiten como también el valor β (igual que en la Ecuación (1)) el cual permite muestrear nuevos individuos con una diversidad similar al mejor valor fitness. De Anda-Suárez et al. en su artículo titulado “*Symmetric-Approximation Energy-Based Estimation of Distribution (SEED): A Continuous Optimization Algorithm*” [7] emplearon el valor de $\beta = 1/g_{x^*}$, en donde g_{x^*} es la mejor solución de la población.

$$v \approx \frac{\frac{1}{Z\beta} \sum_i^m \exp(\beta g_{x_i}) (x_i - \mu)^2 + \sum_i^m g_{x_i} (x_i - \mu)^2}{\sum_i^m g_{x_i}} \quad (3)$$

SEED es un algoritmo iterativo (ver pseudocódigo en Algoritmo 2), el cual consiste en la creación de una población de soluciones candidatas (individuos) creada de manera aleatoria, evaluadas y comparadas con un segundo grupo de individuos. La creación del segundo grupo se realiza mediante la selección de un individuo de la primera población

(ver algoritmo 3), del cual se obtiene su media y desviación estándar mediante la Ecuación (1) y la Ecuación (3), a partir de estas medidas (μ y ν) se crea la segunda población de individuos [7].

Algoritmo 2 Pseudocódigo de SEED

Entrada: N : tamaño de población , n : dimensión , R : rango

Salida: $P_{mejor}^{(t)}$: mejor solución

- 1: $t \leftarrow \text{iteración}$
 - 2: $P^{(t)} \leftarrow$ Inicializar población $\sim U(N, R)$
 - 3: $g(\vec{x}_i) \leftarrow$ Evaluar población ($P^{(t)}$)
 - 4: $P_{mejor}^{(t)} \leftarrow$ obtener la mejor población ($P^{(t)}$)
 - 5: **mientras** el criterio de paro sea falso **hacer**
 - 6: $S^{(t)} \leftarrow$ elegir población ($t, g(\vec{x}_i)$) ver algoritmo 3.
 - 7: obtener $\vec{\mu}^{(t)}$ y $\vec{\nu}^{(t)}$ usando las Ecuaciones (1) y (3)
 - 8: $P^{(t+1)} \leftarrow$ crear nueva población $\sim N(\vec{\mu}^{(t)}, \vec{\nu}^{(t)})$
 - 9: $g(\vec{x}_i) \leftarrow$ evaluar nueva población ($P^{(t+1)}$)
 - 10: $P_{mejor}^{(t+1)} \leftarrow$ obtener la mejor solución ($P^{(t+1)}$)
 - 11: **si** $P_{mejor}^{(t)} \geq P_{mejor}^{(t+1)}$ **entonces**
 - 12: agregar $P_{mejor}^{(t)}$ a $P^{(t+1)}$
 - 13: **fin si**
 - 14: $t \leftarrow t + 1$
 - 15: $P_{mejor}^{(t)} \leftarrow$ obtener las mejores soluciones ($P^{(t)}$)
 - 16: **fin mientras**
 - 17: **devolver** $P_{mejor}^{(t)}$
-

Ambas poblaciones de individuos son evaluadas (obtener el valor fitness de cada individuo) para elegir los mejores individuos que conformarán la primera población de la siguiente iteración, por cada iteración se repetirá el proceso para la creación de una segunda población de individuos que será comparada con la primera población, el criterio de paro de este algoritmo es mediante un número definido de iteraciones.

Algoritmo 3 Pseudocódigo del método de selección

Entrada: t : iteraciones, $g(\vec{x}_i)$: valor fitness de la población

- 1: **si** $t = 0$ **entonces**
 - 2: $\theta^{(t)} = \min\{g(\vec{x}_i)\}$
 - 3: **si no**
 - 4: $g_{min} = \min\{g(\vec{x}_i), \text{tal que } g(\vec{x}_i) \geq \theta^{(t-1)}\}$
 - 5: $g_{med} = g(\vec{x}_{N/2})$
 - 6: $\theta^{(t)} = \max\{g_{min}, g_{med}\}$
 - 7: **fin si**
 - 8: $S^{(t)} \leftarrow$ selecciona la mejor solución donde $g(\vec{x}_i) \geq \theta^{(t)}$
 - 9: **devolver** $S^{(t)}$
-

Descriptor: También conocido como extractor de características permite obtener las propiedades o atributos que identifican a una imagen. En este proceso el identificador resultante de la imagen es conocido como vector de características, este vector está formado por una serie de valores numéricos que representan las propiedades de una imagen [26].

En el estado del arte se pueden encontrar una gran cantidad de descriptores enfocados a la caracterización de imágenes con diferentes niveles de complejidad, a continuación, se mencionan algunos descriptores (Histograma de Gradientes Orientados, histograma de la imagen y Patrones Locales Binarios) que se aplicaron a Evolución Gramatical.

Histograma de Gradientes Orientados (HOG) es un descriptor que utiliza la magnitud y la orientación del gradiente en porciones localizadas de una imagen denominadas “*cel-das*”; estas cantidades son concatenadas construyendo un histograma de una dimensión tomando en cuenta los niveles de gris de los pixeles y usando las ecuaciones (4) y (5), en donde G_x representa la orientación del gradiente en el eje de las x mientras que G_y representa la orientación del gradiente en el eje de las y [9].

$$\text{Magnitud} : |\Delta f| = \sqrt{G_x^2 + G_y^2} \quad (4)$$

$$\text{Orientación} : \theta = \tan^{-1}\left(\frac{G_y}{G_x}\right) \quad (5)$$

El histograma representa la frecuencia relativa de ocurrencia de los niveles de gris en una imagen [24], los niveles de gris se encuentran dentro del rango de 0 y 255. El histograma nos proporciona una vista global de la composición de la imagen y también se puede utilizar como un descriptor, el cual proporciona un vector de características de dimensión 256.

Patrones Locales Binarios (LBP) es un descriptor invariante ante cambios de iluminación en las imágenes, el cual es comúnmente usado debido a que no posee mucha complejidad computacional, al contrario, es un descriptor sencillo de aplicar. LBP consiste en la obtención de una ventana de \mathbf{P} niveles de intensidad en un radio \mathbf{R} , a partir de estos datos se aplica la ecuación (6), en donde g_c indica el nivel de intensidad del pixel central de la ventana, g_p indica el nivel de intensidad del pixel vecino (o que rodea al pixel central) [18].

$$LBP_P^R(X_c, Y_c) = \sum_{p=0}^{p-1} s(g_p - g_c) 2^p \quad (6)$$

$$s(x) = \begin{cases} 1, & \text{if } (x \geq 0) \\ 0, & \text{if } (x < 0) \end{cases}$$

Clasificador: Después de la obtención de los vectores de atributos más significativos

de las imágenes, estos datos entran al proceso de clasificación, el cual consiste en agrupar los vectores en N clases mediante el análisis de la similitud entre los mismos [25]. El análisis de los vectores se realiza dividiéndolos en dos grupos (grupo de entrenamiento y grupo de prueba). El primer grupo se utiliza para el entrenamiento del clasificador, el cual consiste en desarrollar las reglas de decisión del clasificador para obtener el conocimiento acerca del problema que se requiera abordar, de esta manera se encontrará el correcto funcionamiento del clasificador que permita indicar la clase a la que pertenece cada uno de los vectores que se encuentran en el grupo de prueba [24]. El grupo de prueba como ya se mencionó anteriormente será el que indique el rendimiento del clasificador mediante una medida, la cual se puede basar en el error de clasificación (indicando cuántos vectores del grupo de prueba fueron agrupados erróneamente) o en la precisión de clasificación (indicando cuántos vectores del grupo de prueba fueron clasificados correctamente), de esta manera es posible comparar el rendimiento entre clasificadores.

Uno de los clasificadores que destaca por su simpleza en el ámbito de reconocimiento de patrones es el algoritmo K vecinos más cercanos, el cual ha demostrado obtener resultados competitivos ante problemas linealmente separables y los que no son linealmente separables, las máquinas de vector soporte puede ser usado para problemas linealmente separables, no linealmente separables y para problemas de regresión, con ayuda de un kernel que permite la búsqueda de un hiperplano capaz de hacer la distinción entre clases [44]. Ambos clasificadores fueron empleados con evolución Gramatical.

El algoritmo K vecinos más cercanos (KNN por sus siglas en inglés) es un clasificador supervisado (se conoce a qué clase pertenece cada uno de los vectores de atributos), el cual

obtiene la similitud entre los vectores del entrenamiento y un vector de prueba. Posterior a la obtención de las similitudes entre el vector de prueba (al que queremos conocer su clase) y todos los vectores de entrenamiento se obtienen las k clases (por conveniencia se selecciona valor de k impar) de los vectores de entrenamiento con mayor similitud al vector de prueba y se selecciona la clase que sea mayoría [39].

La máquina de vector soporte (SVM por sus siglas en inglés) es un clasificador que se basa en la creación de un hiperplano que separe dos clases de n dimensiones [39] (ver figura 6). El margen funcional es la distancia que separa al hiperplano de los datos de entrenamiento más cercanos al hiperplano (vectores de soporte), la confianza de los resultados de clasificación se basa en la amplitud del margen, entre mayor sea el margen, la confianza mejorará.

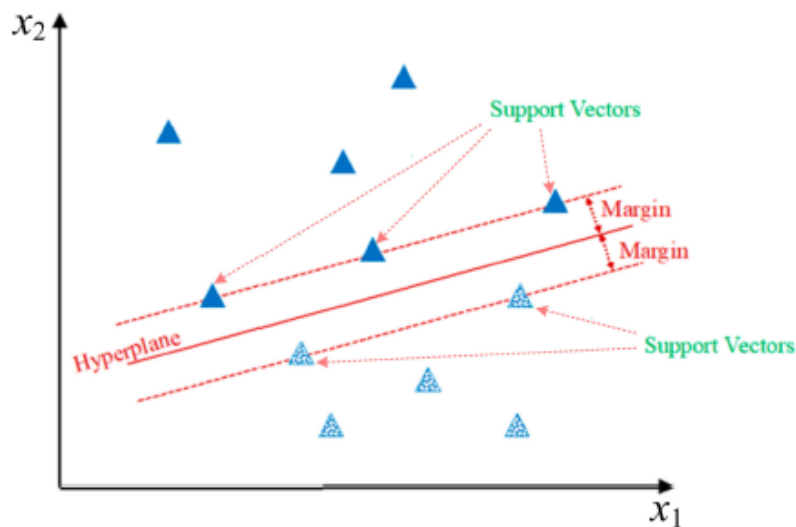


Figura 6: Hiperplano de SVM [39].

En ocasiones los datos no son linealmente separables por lo que se recurre a una función kernel la cual permite medir la similitud entre los vectores de un espacio de mayores dimensiones sin la necesidad de transformar los datos, este espacio es conocido como el espacio de Hilbert y permite la creación de un hiperplano que permita una mejor separabilidad entre clases. Las funciones kernel más conocidas son: lineal, polinomial, base radial y sigmoideal [13].

4. Metodología

Este capítulo inicia con el proceso que se llevó a cabo en la etapa de generación de descriptores y la etapa de evaluación de descriptores. La etapa de generación de descriptores consistió en la obtención del descriptor fenotípico mediante Evolución Gramatical, la etapa de evaluación de descriptores consistió en aplicar el descriptor fenotípico a la base de datos y clasificarla mediante SVM o KNN.

4.1. Etapa de generación de descriptores

La obtención automática de descriptores para la caracterización de imágenes de texturas mediante GE se muestra en la Figura 7. Este método se adaptó para la caracterización y clasificación de expresiones faciales mediante el uso de diferentes bases de datos. Se realizó de esta manera con el fin de comprobar si GE puede ser aplicado a distintos problemas dentro del reconocimiento de imágenes con resultados aceptables y comparables a otras metodologías.

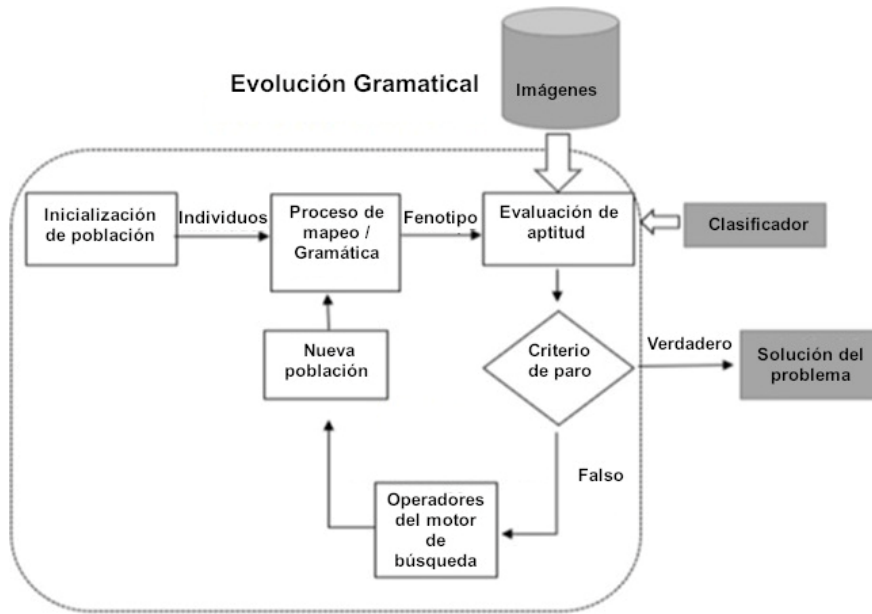


Figura 7: Diagrama de generación de descriptores [5].

Las entradas requeridas por GE fueron: el conjunto de imágenes de las bases de datos (JAFFE, CK+ y KDEF), un motor de búsqueda y un algoritmo clasificador. Inicialmente se creó una población de individuos llamados cromosomas con una determinada dimensión y con valores numéricos enteros aleatorios creados a partir de una distribución uniforme. Por medio de la gramática BNF y el proceso de mapeo Depth-First, los individuos se transformaron a su forma fenotípica y su calidad fue evaluada por una función de aptitud. Luego, los operadores del motor de búsqueda fueron aplicados a la población para obtener nuevos individuos que puedan mejorar su desempeño en el problema. Este proceso se repitió hasta que se cumplió el criterio de paro (número de generaciones).

DE fue uno de los dos motores de búsqueda que se aplicó en el esquema de GE, en el cual los operadores proporcionados por el motor de búsqueda fueron: mutación (em-

pleando un coeficiente de muta), cruza (empleando un coeficiente de cruza) y selección. El segundo motor de búsqueda usado fue SEED cuyos operadores aplicados a la gramática fueron los siguientes: obtención de la media (μ), obtención de la desviación estándar (ν), remuestreo de población y selección.

Gramática BNF. Los descriptores fenotípicos generados por GE se formaron por la combinación de un conjunto de operadores básicos agrupados en filtros pasa-alta, pasa-baja, direccionales, Max-Pooling, escala de grises y operadores aritméticos [5]. Estos operadores constituyen los símbolos terminales de la gramática BNF. En la Tabla 2 se muestra una descripción de estos operadores y sus correspondientes símbolos empleados en la gramática.

Tabla 2: Operadores básicos utilizados en la Gramática BNF [5].

Tipo de operador	Nombre	Simbolo terminal en la Gramática
Pasa-alta	Laplaciano	Lap
Pasa-alta	Laplaciano de la Gaussiana	LapG1, LapG2
Pasa-alta	Derivada de la Gaussiana	GauDX, GauDY
Pasa-baja	Gaussiana	Gau1, Gau2
Pasa-baja	Media	AverF
Pasa-baja	Mediana	MedianF
Direccional	Banco de Filtros de Gabor	GFB0, GFB45, GFB90, GFB135
Max-Pooling	Max Pooling	MP2, MP4, MP6, MP8, MP10
Histograma	Ecualización del Histograma	HEq
Operador aritmético	Valor Absoluto de una imagen	AbsV
Operador aritmético	Cuadrado de una imagen	Sqr
Operador aritmético	Raíz cuadrada de una imagen	Sqrt
Operador aritmético	Logaritmo base 2	Log
Operador aritmético	Multiplicar una imagen por 0.5	T0.5
Operador aritmético	Sumar 2 imágenes	ADD
Operador aritmético	Restar 2 imágenes	SUB
Operador aritmético	Resta absoluta de 2 imágenes	ASUB
Operador aritmético	Multiplicar 2 imágenes	MUL
Operador aritmético	Dividir 2 imágenes	DIV
Escala de grises	Obtener imagen a escala de grises	lg

Los símbolos no terminales empleados en la gramática BNF fueron los siguientes:

$N = \langle Start \rangle, \langle MP \rangle, \langle Expr \rangle, \langle Filter \rangle, \langle Gau \rangle, \langle Lap \rangle, \langle GFB \rangle, \langle Arith \rangle, \langle Op \rangle, \langle Terminal \rangle$

Las reglas de producción se muestran en la Tabla 3, las cuales son separadas por el símbolo ”|”. Por lo tanto, un símbolo no terminal puede tener una o varias reglas de producción, las cuales se enumeran para el proceso de mapeo. El símbolo inicial es $\langle Start \rangle$.

Tabla 3: Reglas de producción utilizadas en la Gramática BNF [5].

Símbolo no terminal	Reglas de producción
$\langle Start \rangle$	$\langle Expr \rangle \langle MP \rangle (0)$
$\langle Expr \rangle$	$\langle Expr \rangle \langle Expr \rangle \langle Op \rangle (0) \mid \langle Expr \rangle \langle Filter \rangle (1)$ $\mid \langle Terminal \rangle (2)$
$\langle Op \rangle$	$ADD (0) \mid SUB (1) \mid ASUB (2) \mid MUL (3) \mid DIV (4)$
$\langle Filter \rangle$	$\langle Gau \rangle (0) \mid \langle Lap \rangle (1) \mid \langle GFB \rangle (2) \mid \langle Arith \rangle (3)$
$\langle Gau \rangle$	$Gau1 (0) \mid Gau2 (1) \mid GauDX (2) \mid GauDY (3)$
$\langle Lap \rangle$	$LapG1 (0) \mid LapG2 (1) \mid Lap (2)$
$\langle GFB \rangle$	$GFB0 (0) \mid GFB45 (1) \mid GFB90 (2) \mid GFB135 (3)$
$\langle Arith \rangle$	$AverF (0) \mid MedianF (1) \mid HEq (2) \mid AbsV (3) \mid Sqr (4) \mid Sqrt (5)$ $\mid Log (6) \mid T0.5 (7)$
$\langle MP \rangle$	$MP2 (0) \mid MP4 (1) \mid MP6 (2) \mid MP8 (3) \mid MP10 (4)$
$\langle Terminal \rangle$	$lg (0)$

Función de aptitud. La calidad de los fenotipos (descriptores) generados por GE se evaluó de la siguiente manera:

- 1) Aplicación el fenotipo a la base de datos. Para esto el descriptor fenotípico se aplicó a cada una de las imágenes de la partición de la base de datos designada para obtener un nuevo grupo de imágenes procesadas.
- 2) Obtención de los vectores de características. Al conjunto de imágenes obtenidas en el punto anterior se le aplicó un descriptor seleccionado (HOG, histograma o LBP) para así obtener su vector de características.
- 3) Clasificación. Los vectores de características obtenidos en el punto anterior fueron etiquetados de acuerdo a su clase perteneciente y clasificados usando SVM o KNN. Así se obtuvo el porcentaje de clasificación que representó el valor de aptitud del fenotipo.

Es deseable que el valor de aptitud de los fenotipos se aproxime al 100 %, por lo que a la función se le consideró como un problema de maximización.

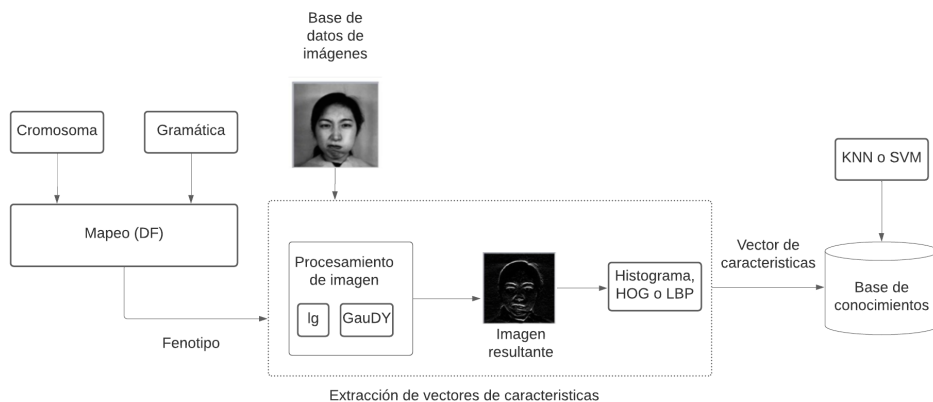


Figura 8: Diagrama de extracción de características [5].

Los puntos mencionados anteriormente para la evaluación de la aptitud de un fenotipo se muestran en el diagrama de la Figura 8, el cual originalmente fue empleado en imágenes de texturas y fue adaptado al reconocimiento de expresiones faciales. En donde el cromosoma se convirtió en un fenotipo (por ejemplo: “*lg, GauDY*”) empleando la gramática y un proceso de mapeo, posteriormente para la extracción de los vectores de características el fenotipo fue aplicado a la base de datos de imágenes para la obtención de un nuevo grupo de imágenes resultantes, estas imágenes fueron transformadas en vectores de características mediante un descriptor (Histograma, HOG o LBP), cada uno de estos vectores fue almacenado en la base de conocimientos para que posteriormente fueran clasificados por SVM o KNN.

La razón por la se emplearon distintos clasificadores (KNN y SVM), descriptores (HOG, Histograma y LBP) y motores de búsqueda (DE y SEED) fue para la búsqueda

de mejores resultados dentro de GE, debido a que no se sabe cuáles herramientas son las más favorables para la obtención del mejor descriptor fenotípico. Se seleccionaron los clasificadores (KNN y SVM) debido a que son empleados en el estado del arte para el reconocimiento de expresiones faciales. Los descriptores LBP y el histograma de una imagen son descriptores sencillos y computacionalmente baratos por lo que fueron apropiados para implementarse con GE, lo que permitió realizar un mayor número de generaciones en un menor tiempo de cómputo. HOG fue un descriptor que representó un mayor costo computacional, pero en el estado del arte demostró tener buenos resultados en el ámbito de reconocimiento de expresiones faciales. Los motores de búsqueda se encargan del proceso evolutivo dentro de GE, lo que permite la obtención de distintos descriptores fenotípicos. En Calzada et al. [5] se implementó DE como motor de búsqueda para imágenes de texturas y obtuvo resultados adecuados, por lo que se tomó en cuenta para el ámbito de reconocimiento de expresiones faciales, SEED al emplear modelos probabilísticos para la búsqueda de mejores resultados dentro del un espacio de muestra, resultó ser una alternativa diferente a la búsqueda que emplea DE.

A continuación, se muestra un ejemplo de un descriptor fenotípico generado por GE para una imagen de la base de datos JAFFE (ver Figura 9). El descriptor generado es “*lg , GauDY , lg , SUB , Gau2 , MP6*” que mediante notación postfija se fue aplicando paso por paso y de izquierda a derecha a la imagen de prueba, obteniendo una imagen procesada. A la imagen resultante se le aplicó HOG para obtener un vector de características. Este proceso se repite para cada una de las imágenes de la base de datos JAFFE.

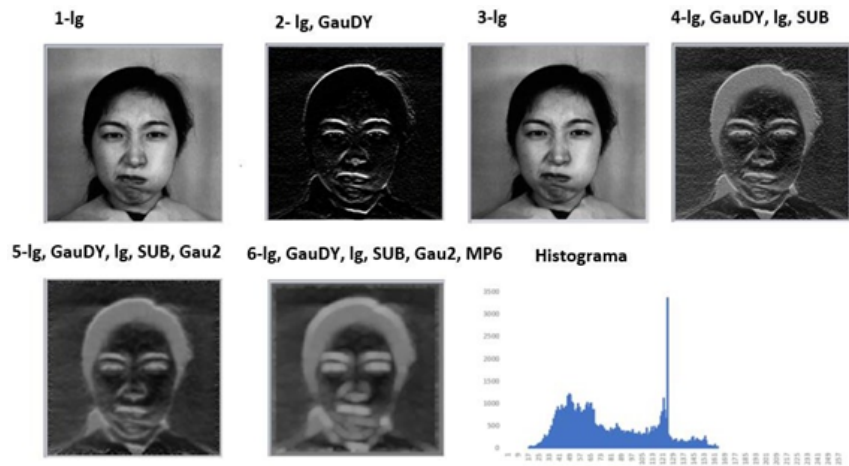


Figura 9: Ejemplo del proceso de caracterización de una imagen.

4.2. Etapa de evaluación de descriptores

El mejor descriptor fenotípico obtenido en la etapa de generación de descriptores fue aplicado a la partición de la base de datos designada a la evaluación de descriptores. Posteriormente, se obtuvieron los vectores de características de las imágenes (usando HOG, histograma o LBP) para que fueran clasificados y de esta manera se obtuvo el porcentaje de clasificación.

5. Experimentación y resultados

En esta sección se muestran los experimentos realizados y los resultados obtenidos.

5.1. Bases de datos

Japanese Female Facial Expressions (JAFFE) fue una de las bases de datos de imágenes de expresiones faciales que se emplearon [23], la cual contiene 213 imágenes de rostros de 10 mujeres japonesas, agrupadas en 6 expresiones faciales básicas además de una expresión neutra, quedando distribuidas como sigue: 30 imágenes neutrales, 30 imágenes de enojo, 29 imágenes de disgusto, 32 imágenes de miedo, 31 imágenes de alegría, 31 imágenes de tristeza y 30 imágenes de sorpresa. Cada imagen tiene una dimensión de 256×256 píxeles.

Extended Cohn-Kanade (CK+) contiene 520 imágenes de rostros de 210 adultos entre 18 y 50 años de edad, de los cuales 69% corresponden a mujeres, 81% a personas euro-americanas, 13% afro-americanos y un 6% corresponden a otros grupos de personas [21]. Al igual que la base de datos JAFFE, CK+ esta agrupado por 6 expresiones faciales básicas y una neutra, el tamaño original de las imágenes es de 640×480 píxeles y se redujo a 128×128 empleando el algoritmo de cascadas Haar [40].

Karolinska Directed Emotional Faces (KDEF) es un grupo de 4900 imágenes de expresiones faciales de 70 individuos distribuidas en 7 emociones diferentes tomadas en 5 ángulos, en dónde el tamaño de cada imagen fue de 562×762 píxeles. Los individuos fueron 70 actores novatos (35 hombres y 35 mujeres) cuya edad se encuentra en el rango

de 20 a 30 años [22]. De las 4900 imágenes únicamente 980 fueron tomadas de frente, esas imágenes fueron el resultado de 2 tomas idénticas por persona por lo que se decidió tomar únicamente la primera toma quedando 490 imágenes. El algoritmo cascadas Haar se usó para redimensionar las imágenes a un tamaño de 256×256 .

5.2. Comparativa entre clasificadores

Evolución Gramatical al ser una metodología en la cual la obtención de un descriptor que nos pueda proporcionar un porcentaje de clasificación alto dependerá de la cantidad de generaciones o ciclos generados, esto implica grandes costos computacionales (tiempo de ejecución) por lo que se realizó una comparativa previa a la implementación de Evolución Gramatical entre los clasificadores K-NN y SVM.

Se tomó como antecedentes los resultados obtenidos por Cornejo [6] mencionados en el estado del arte, en donde los porcentajes de clasificación promedio se inclinaron hacia el clasificador SVM. Se realizó la implementación de HOG con la base de datos JAFFE empleando SVM y K-NN (con $k=1$), los parámetros empleados por SVM fueron: Kernel lineal, valor $C=1$ tolerancia de 0.001 y validación cruzada. Los parámetros de HOG fueron los siguientes: Redimensionamiento de imagen a 128×128 , celdas de 16×16 , 2 bins. En la Tabla 4 se pueden observar los resultados que se obtuvieron tomando en cuenta validación cruzada con diferentes cantidades de pliegues los cuales indicaron que el clasificador SVM de 10 pliegues fue el que obtuvo un porcentaje de clasificación mayor (83.87%).

Tabla 4: Resultados de JAFFE y HOG.

Clasificador	Folds	Porcentaje de Clasificación
SVM	3	80.64 %
SVM	5	82.02 %
SVM	10	83.87 %
KNN	3	70.04 %
KNN	5	80.18 %
KNN	10	81.10 %

A partir de los resultados ya mencionados se tomó la decisión de implementar Evolución Gramatical empleando únicamente el clasificador SVM con los parámetros mencionados anteriormente y validación cruzada de 10 pliegues.

5.3. Esquemas de experimentación

5.3.1. Primera experimentación

El primer esquema de experimentación consistió en utilizar la base de datos de imágenes completa para la generación de descriptores fenotípicos. De esta manera se proporcionó la información completa de toda la base de datos al proceso evolutivo de GE con la finalidad de obtener los mejores resultados.

Los valores de los parámetros utilizados fueron los siguientes: El tamaño de la población fue de 20 cromosomas, donde cada cromosoma estuvo formado por 20 codones con valores enteros en el rango de $[0, 255]$. El proceso evolutivo consistió de 80 generaciones. El motor de búsqueda usado en este esquema de experimentación fue DE en combinación

con HOG, SVM y la base de datos JAFFE (imágenes de 256×256 píxeles). Los parámetros pertenecientes a HOG fueron: celdas de 32×32 píxeles y 5 bins. En el proceso de clasificación para la obtención del valor de aptitud de los fenotipos, se utilizó SVM con un kernel lineal, con $C = 1$, una tolerancia de 0.001 y validación cruzada con 10 pliegues. Por último, DE empleó un coeficiente de mutación de 0.5 y el coeficiente de cruce de 0.8.

A continuación, en la Tabla 5 se muestra una comparativa entre los resultados obtenidos en esta experimentación y otros resultados del estado del arte aplicados a la base de datos JAFFE. El resultado reportado en nuestra propuesta es el promedio de 35 experimentos independientes, en donde para cada experimento se tomó en cuenta el porcentaje de clasificación más alto de los resultados obtenidos en la última generación.

Tabla 5: Comparación del desempeño de diferentes metodologías usando la base de datos JAFFE.

Autor	Metodología	Porcentajes de Clasificación
Eng. et al.[9]	HOG, SVM	88.89 %
Yang et al. [43]	Banco de Filtros de Gabor + Descriptor Local de Weber, KNN	92.77 %
Pitaloka et al. [31]	Deep Learning	97.06 %
Melaugh et al. [27]	Red Neuronal Convolutacional	76.56 %
Nuestra Propuesta	Evolución Gramatical + HOG, SVM	97.78 %

Los resultados de la Tabla 5 indican que Evolución Gramatical obtuvo resultados comparables a otras metodologías de caracterización como Deep Learning e indican que superó el resultado de descriptores que son estándar en el reconocimiento de imágenes (Banco de Filtros de Gabor). Una de las principales aportaciones de estos resultados fue

que se pudo observar en cierto grado la mejoría del porcentaje de clasificación obtenido empleando Evolución Gramatical, en donde en el artículo de Eng et al. emplearon el mismo clasificador (SVM) y el descriptor HOG y obtuvieron un porcentaje de clasificación del 88.89% mientras que Evolución Gramatical alcanzó un 97.78%.

A continuación se realizaron pruebas tomando en cuenta toda la base de datos para la etapa de generación de descriptores con los distintos motores de búsqueda (DE y SEED) aplicados a las bases de datos de imágenes JAFFE (256×256 pixeles) y CK+ (128×128 pixeles) y usando solamente HOG como descriptor. Los valores de los parámetros fueron los siguientes: tamaño de la población fue de 10 cromosomas, en donde cada cromosoma estuvo formado por 10 codones con valores enteros de $[0,255]$. El proceso evolutivo consistió en 10 generaciones. Los parámetros que empleó DE fueron: coeficiente de cruce de 0.8 y coeficiente de muta de 0.5. Para el descriptor HOG y el algoritmo SVM se emplearon los mismos parámetros que en el primer esquema de experimentación. Se realizó un experimento con cada motor de búsqueda y base de datos, utilizando validación cruzada con 10 pliegues. En la Tabla 6 se reporta el mejor fitness de la última generación, en donde se observa que los resultados obtenidos con la base de datos CK+ están muy por debajo de los reportados en el estado del arte Jain et al. [16] 85.32%, Turan y Lam [38] 95.90%, Mohan et al. [28] 96.79%, Paraneswaran et al. [30] 91%.

Tabla 6: Mejor fitness de JAFFE y CK+ con HOG y distintos motores de búsqueda.

Base de Datos	Motor de Búsqueda	Mejor Fitness
JAFFE	SEED	97.19 %
JAFFE	DE	96.25 %
CK+	SEED	66.20 %
CK+	DE	65.26 %

5.3.2. Segunda experimentación

Debido a los bajos resultados mostrados en la Tabla 6 con CK+, y previo a un segundo esquema de experimentación, se diseñó otro procedimiento, el cual consistió en separar las imágenes con emoción neutra del resto de las imágenes y se les aplicó a ambos grupos el descriptor fenotípico generado y el descriptor designado (HOG, LBP o histograma). Tras la obtención de los dos grupos de vectores de características (vectores de imágenes de emociones neutras y vectores de imágenes del resto de emociones), se calculó el vector final para cada individuo de la base de datos mediante la siguiente formula: $\vec{V}_f = |\vec{V}_n - \vec{V}_e|$.

En donde \vec{V}_f : es el vector final, \vec{V}_n : es el vector del individuo con emoción neutra y \vec{V}_e : es el vector del individuo con cualquier emoción excepto la neutra. El vector final alimentó al clasificador SVM. Esta metodología se aplicó únicamente a la base de datos CK+ en el segundo esquema de experimentación.

Con el objetivo de conocer el desempeño de los mejores descriptores fenotípicos generados por GE con imágenes desconocidas en el proceso evolutivo, se realizó el segundo esquema de experimentación, donde se dividió la base de datos en 2 subconjuntos, uno

para la generación evolutiva de los descriptores y el otro para la evaluación del mejor descriptor generado. Se decidió formar cada subconjunto con el 50% de las imágenes de la base de datos, debido a que el proceso de GE requiere bastantes recursos computacionales a medida que aumenta el número de imágenes utilizadas.

En GE se emplearon 2 motores de búsqueda (DE y SEED), 3 bases de datos de imágenes (JAFFE, CK+ y KDEF) y el clasificador SVM con los mismos parámetros que en el primer esquema de experimentación.

En el diseño evolutivo de los descriptores se realizaron 15 experimentos independientes con cada base de datos, con su respectivo descriptor (HOG, LBP y el histograma) y por cada motor de búsqueda. Los parámetros empleados para estos experimentos fueron los dados en la Tabla 7.

Tabla 7: Parámetros del segundo esquema de experimentación.

Descriptores	JAFFE	CK+	KDEF
Histograma	_____	_____	_____
LBP	8 vecinos, radio 1	8 vecinos, radio 1	8 vecinos, radio 1
HOG	Celdas 128×128 , bloques 32×32 , 5 bins	Celdas 16×16 , bloques 16×16 , 2 bins	Celdas 128×128 , bloques 32×32 , 5 bins

Los parámetros utilizados en CK+ con HOG son diferentes debido a que el costo computacional (tiempo) incrementa mucho debido a la metodología que se tomó para CK+ (resta de vectores) por lo mismo, tanto en DE (con coeficiente de mutación de 0.5 y de cruza de 0.8) como en SEED la cantidad de generaciones empleadas en HOG fueron

20, mientras que con el histograma y LBP fueron 50.

Tabla 8: Porcentaje de clasificación promedio en la etapa de generación de descriptores con el segundo esquema de experimentación.

Base de datos / Descriptor	DE	SEED
JAFFE – Histograma	30.03 %	28.10 %
JAFFE - HOG	80.93 %	81.75 %
JAFFE – LBP	27.23 %	26.48 %
CK+ -Histograma	24.47 %	22.71 %
CK+ - HOG	56.34 %	57.86 %
CK+ - LBP	25.30 %	22.29 %
KDEF - Histograma	22.61 %	23.46 %
KDEF – HOG	68.73 %	68.11 %
KDEF – LBP	23.92 %	24.57 %

En la Tabla 8 se muestran los promedios de los resultados de la generación de descriptores fenotípicos de los 15 experimentos en donde se puede observar que el descriptor HOG tiene los mejores resultados tanto en SEED como en DE (marcados en negrita). Se puede observar que los resultados de la Tabla 8 en la base de datos JAFFE en donde se empleó el 50 % del todas las imágenes con el descriptor HOG obtuvo resultado inferiores a los mostrados en el primer esquema de experimentación (Tabla 6), mientras que el primer esquema de experimentación con JAFFE y DE se obtuvo un 96.25 % y con el motor de búsqueda SEED JAFFE obtuvo un 97.19 %, los resultados del segundo esquema obtuvieron un 80.93 % (JAFFE y DE) y un 81.75 % (JAFFE y SEED) respectivamente. En el caso de la base de datos CK+ a pesar de que en el segundo esquema de experimentación se cambio el procedimiento los resultados siguieron siendo inferiores al primer esquema, el primer esquema con un 66.20 % (SEED) y 65.26 % (DE) mientras que el segundo con un 57.86 % (SEED) y 56.34 % (DE) respectivamente.

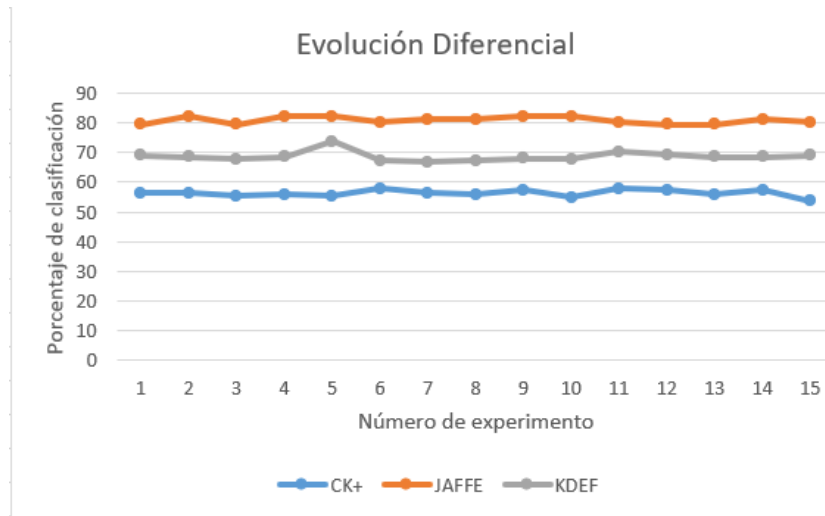


Figura 10: Gráfica de desempeño del descriptor HOG con DE.

En la figura 10 se muestran los mejores resultados en la generación de descriptores de los 15 experimentos utilizando el descriptor HOG y Evolución Diferencial como motor de búsqueda. Se puede observar que la base de datos con la que se obtuvieron los mejores resultados fue JAFFE con una desviación estandar del 1.16%. Los resultados de la base de datos KDEF mostraron una desviación estandar del 1.65%. CK+ fue el que obtuvo resultados más bajos con una desviación estandar del 1.17%.

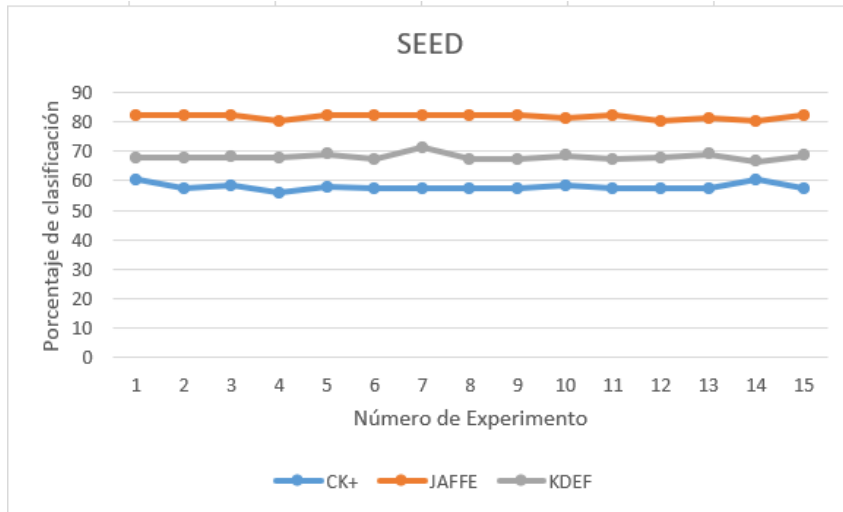


Figura 11: Gráfica de desempeño del descriptor HOG con SEED.

En la figura 11 se muestran los mejores resultados en la generación de descriptores por experimento, en donde se empleó SEED como motor de búsqueda. Las desviaciones estándar de cada base de datos fueron las siguientes: JAFFE 0.79 %, KDEF 1.18 % y CK+ 0.93 %. Los resultados de SEED comparados con DE para JAFFE fueron ligeramente superiores debido a que el menor resultado de SEED fue de 80.38 % comparado con el 79.44 % de DE y el mejor resultado de SEED fue igual que DE de 82.25 %. El menor resultado que obtuvo KDEF con SEED fue de 66.54 % (ligeramente menor del 66.94 % de DE), mientras que el mayor resultado con SEED fue de 71.43 % (por debajo del 73.88 % de DE), por lo que KDEF obtuvo un mejor desempeño con DE. CK+ con SEED obtuvo el peor resultado de 55.99 % y el mejor de 60.29 %, superando a los resultados del motor de búsqueda DE (53.59 % y 57.90 %).

Basándose en los resultados de la Tabla 8 (en donde para cada experimento se tomó en cuenta el mejor valor fitness de la última generación) se seleccionó el descriptor fenotípico

que corresponde a la mediana de los resultados de los 15 experimentos. En las Tablas 9, 10 y 11 se muestran los resultados de aplicar el descriptor fenotípico al segundo conjunto de imágenes (conjunto de imágenes encargados de evaluar el mejor fenotipo) de las bases de datos CK+, JAFFE y KDEF, respectivamente.

Tabla 9: Comparación de resultados de generación y evaluación de descriptores en la base de datos CK+.

Motor de búsqueda/ Descriptor	Porcentaje clasificación (generación de descriptores)	Porcentaje de clasificación (evaluación de descriptores)
DE- Histograma	24.89 %	15.38 %
DE – HOG	56.46 %	57.21 %
DE - LBP	24.89 %	22.11 %
SEED- Histograma	22.01 %	22.11 %
SEED – HOG	57.42 %	59.13 %
SEED - LBP	22.01 %	22.11 %

Tabla 10: Comparación de resultados de generación y evaluación de descriptores en la base de datos JAFFE.

Motor de búsqueda/ Descriptor	Porcentaje clasificación (generación de descriptores)	Porcentaje de clasificación (evaluación de descriptores)
DE- Histograma	29.91 %	21.69 %
DE – HOG	81.31 %	70.75 %
DE - LBP	26.17 %	12.26 %
SEED- Histograma	28.04 %	14.15 %
SEED – HOG	82.25 %	72.64 %
SEED - LBP	26.17 %	12.26 %

Tabla 11: Comparación de resultados de generación y evaluación de descriptores en la base de datos KDEF.

Motor de búsqueda/ Descriptor	Porcentaje clasificación (generación de descriptores)	Porcentaje de clasificación (evaluación de descriptores)
DE- Histograma	22.05 %	15.51 %
DE – HOG	68.58 %	66.53 %
DE - LBP	24.49 %	10.20 %
SEED- Histograma	24.09 %	18.36 %
SEED – HOG	67.76 %	65.71 %
SEED - LBP	24.09 %	16.73 %

En la Tabla 9 se puede observar que únicamente con el descriptor HOG aplicado a la base de datos CK+, los resultados de evaluación del fenotipo fueron superiores a los de generación de los descriptores fenotípicos (marcados en negrita); mientras que en los demás casos, incluyendo los de las Tablas 10 y 11, sucedió lo contrario. Creemos que esto se debe a que en el proceso de diseño de los descriptores fenotípicos, Evolución Gramatical va adquiriendo conocimiento del grupo de imágenes de generación de descriptores.

A partir de los resultados mostrados en las tablas anteriores, se puede observar que HOG tuvo un mejor desempeño que el histograma y LBP. Esto puede ser debido a que HOG se basa en la orientación de los gradientes de la imagen, mientras que el histograma y LBP se basan solamente en los niveles de gris. Por lo que se puede suponer que para el caso de las bases de datos JAFFE, CK+ y KDEF la orientación del gradiente proporciona mejores características de las imágenes.

También de las tablas anteriores se observa que al aplicar el descriptor HOG a las

bases de datos, JAFFE obtuvo los mejores resultados, seguida de KDEF y al final CK+. Creemos que CK+ tiene los resultados de clasificación más bajos debido al tamaño de las imágenes (128×128 píxeles), mientras que las imágenes de JAFFE y KDEF tienen un tamaño de 256×256 píxeles, lo cual implica mayor información de la imagen. Así también cabe mencionar que la base de datos JAFFE es la que tiene la mejor uniformidad y nivel de iluminación entre sus imágenes, a comparación de las otras 2 bases de datos, lo cual también podría influir en los resultados de clasificación (ver Figura 12).

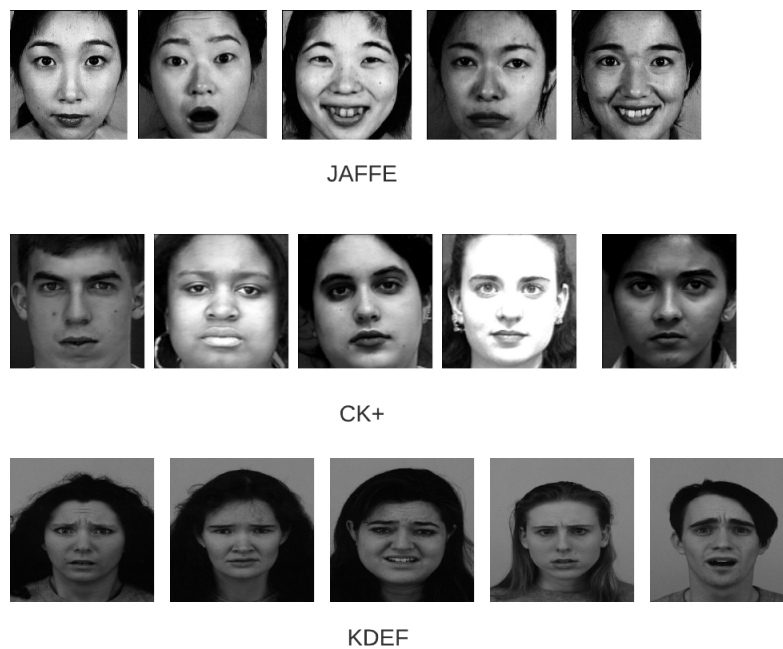


Figura 12: Ejemplos de imágenes de las bases de datos JAFFE, CK+ y KDEF.

Tabla 12: Resultados de clasificación sin utilizar Evolución Gramatical.

Descriptor-Base de datos	Porcentaje de clasificación (generación de descriptores)	Porcentaje de clasificación (evaluación de descriptores)
Histograma - JAFFE	22.42 %	9.43 %
Histograma - CK+	15.31 %	13.46 %
Histograma - KDEF	11.02 %	11.02 %
HOG - JAFFE	79.43 %	70.75 %
HOG - CK+	53.11 %	60.09 %
HOG - KDEF	66.53 %	66.12 %
LBP - JAFFE	15.88 %	16.98 %
LBP - CK+	16.74 %	20.67 %
LBP - KDEF	20.40 %	9.79 %

Para comparar el desempeño de Evolución Gramatical con el de los descriptores HOG, LBP e Histograma aplicados en forma individual, se realizaron pruebas con estos descriptores usando las mismas bases de datos (50% de generación y 50% de evaluación de descriptores fenotípicos) y el algoritmo SVM para clasificación con los mismos parámetros usados en GE y con validación cruzada de 10 pliegues. Los resultados obtenidos se muestran en la Tabla 12, donde se puede observar que todos los porcentajes de clasificación obtenidos con los grupos de generación de descriptores fenotípicos fueron inferiores a los obtenidos por GE. Los resultados obtenidos con el subconjunto de imágenes de las bases de datos encargados de la evaluación de los mejores fenotipos, solamente fueron superiores a los alcanzados con GE empleando HOG con CK+ (60.09%) y LBP con JAFFE (16.98%) (anotados en negrita en la Tabla 12).

5.3.3. Análisis de ocurrencia de operadores en GE

Finalmente, se realizó un análisis en cuanto a qué descriptores (filtros) básicos fueron más aptos durante el proceso de Evolución Gramatical. Para ello se realizó el cálculo de la frecuencia con que apareció cada filtro en el mejor descriptor fenotípico de los 15 experimentos independientes. A continuación se muestran los resultados obtenidos.

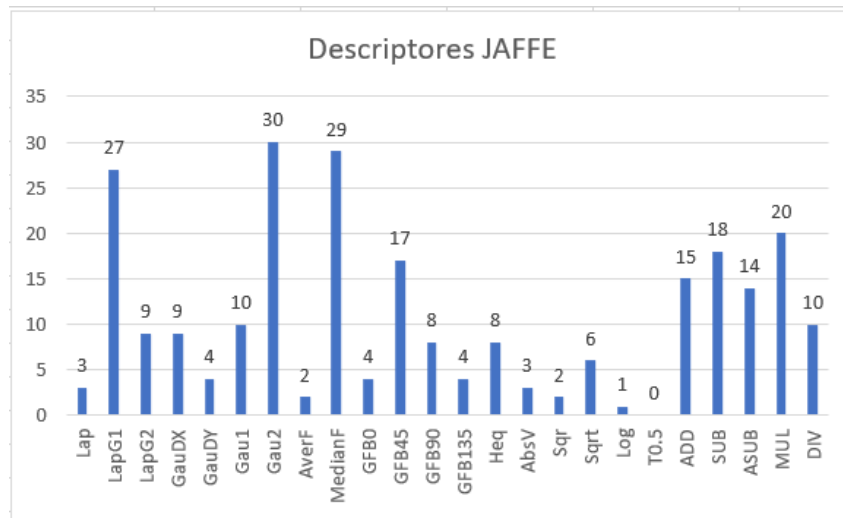


Figura 13: Gráfica de frecuencias de descriptores con la base de datos JAFFE.

En la Figura 13 se muestra la gráfica de frecuencias con que se encontró cada filtro, en donde se puede observar que para la base de datos JAFFE, los Bancos de Filtros de Gabor (*GFB*) fueron poco seleccionados, mientras que otros filtros como la Gaussiana con sigma igual a 2 (*Gau2*), el laplaciano de la Gaussiana (*LapG1*) o el filtro de la mediana (*MedianF*) fueron los que más se utilizaron.

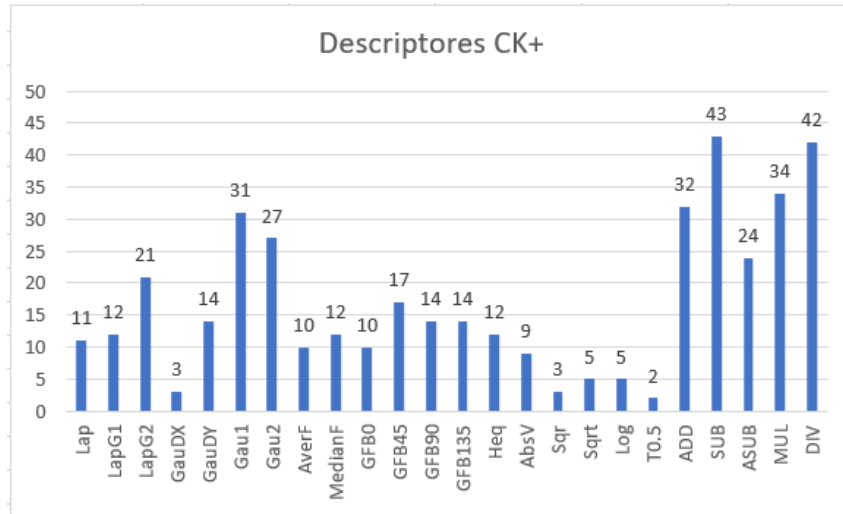


Figura 14: Gráfica de frecuencias de descriptores con la base de datos CK+.

En la Figura 14 se muestra la gráfica de frecuencias de los filtros con la base de datos CK+, en donde a comparación de JAFFE, los Bancos de Filtros de Gabor tuvieron ligeramente mayor utilización pero aún siguen siendo menos frecuentes que la Gaussiana. En el caso de los operadores aritméticos (*ADD*, *SUB*, *ASUB*, *MUL*, *DIV*), debido a que en su funcionamiento intervienen 2 imágenes, permitieron una mayor exploración en el espacio de búsqueda para encontrar un descriptor con mejor desempeño; lo cual implicó que este tipo de operadores fueran de los más utilizados en las bases de datos.

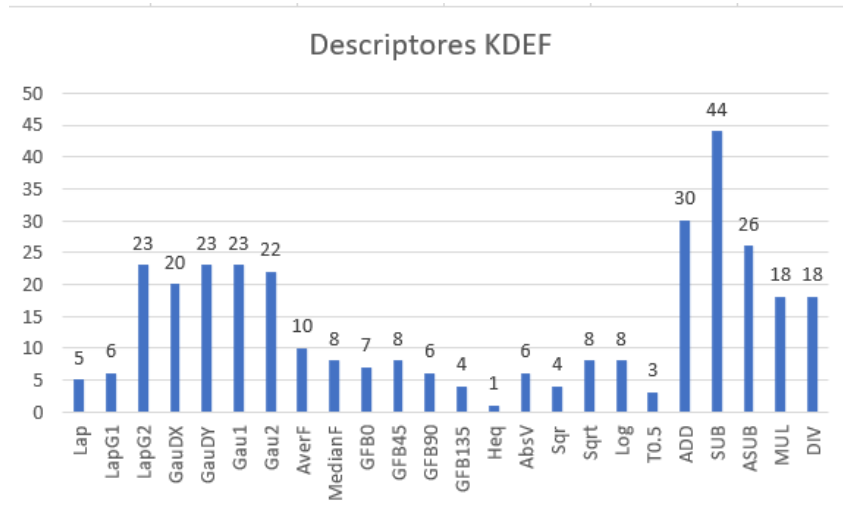


Figura 15: Gráfica de frecuencias de descriptores con la base de datos KDEF.

En la Figura 15 se muestra la gráfica de frecuencias de los filtros con la base de datos KDEF, en donde se observa que los filtros simples fueron los más empleados por la gramática y que los Bancos de Filtros de Gabor fueron más escasos que en las otras bases de datos. Los resultados de las 3 gráficas anteriores se muestran en la Figura 16, en donde se puede observar que el filtro que obtuvo mayor protagonismo en la creación de descriptores fenotípicos fue la resta entre 2 imágenes(*SUB*). Haciendo una comparativa entre los filtros que requirieron únicamente una imagen para su funcionamiento, el filtro de la Gaussiana con valor de sigma igual a 2 fue el que obtuvo mayor protagonismo por encima de los Bancos de Filtros de Gabor y de los laplacianos.

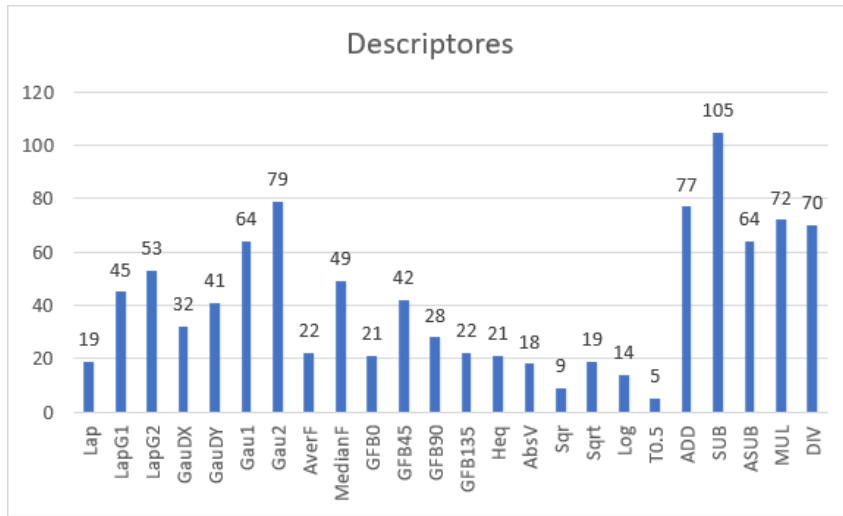


Figura 16: Gráfica de frecuencias de descriptores con las 3 bases de datos.

6. Conclusiones

En esta tesis se propuso una metodología, basada en Evolución Gramatical, para la generación automática de descriptores aplicados al reconocimiento de imágenes de expresiones faciales. En la experimentación se emplearon distintas bases de datos de imágenes de expresiones faciales conocidas en el estado del arte (JAFFE, CK+ y KDEF). Evolución Gramatical utilizó una gramática en la forma Backus-Naur, distintos motores de búsqueda (Evolución Diferencial y SEED) y el proceso de mapeo Depth-First para generar descriptores sintácticamente correctos que se aplicaron a las imágenes de la base de datos. Posteriormente las imágenes procesadas fueron caracterizadas utilizando distintos descriptores (histograma de la imagen, HOG y LBP) y los vectores obtenidos se clasificaron mediante una Máquina de Vector Soporte.

Para evaluar esta metodología con los descriptores y bases de datos mencionados, se diseñaron dos esquemas de experimentación: El primero consistió en tomar en cuenta toda la base de datos de imágenes aplicado a Evolución Gramatical y reportar el mayor porcentaje de clasificación obtenido con una determinada cantidad de generaciones. Este esquema se elaboró tomando en cuenta el motor de búsqueda Evolución Diferencial, la base de datos JAFFE y el descriptor HOG, se realizaron 35 experimentos reportando los mejores resultados obtenidos, en donde el promedio de estos resultados se encontró por encima del 90 %. El segundo esquema de experimentación consistió en dividir las bases de datos en 2 grupos (Generación de descriptores fenotípicos y Evaluación de descriptores fenotípicos) con la misma cantidad de imágenes. El grupo de imágenes para la generación de descriptores se utilizó en Evolución Gramatical para obtener los descriptores fenotípi-

cos, quedándonos con aquellos que tuvieron los mejores porcentajes de clasificación en la etapa de diseño. El descriptor correspondiente a la mediana de los mejores resultados de 15 experimentos independientes se aplicó al grupo de imágenes de evaluación de descriptores y sus vectores de características se clasificaron mediante SVM con validación cruzada a 10 pliegues. De esta manera se obtuvo el desempeño de la segunda metodología implementada.

Los resultados obtenidos en el segundo esquema de experimentación fueron bajos con respecto al primer esquema, debido a que la cantidad de imágenes que se emplearon en el diseño de los descriptores fenotípicos fue reducida a la mitad. Sin embargo, estos resultados fueron mejores a los obtenidos por los descriptores HOG, LBP e Histograma aplicados en forma individual (sin proceso evolutivo) a las mismas bases de datos.

Evolución Gramatical, por ser una metodología genérica, nos permitirá incorporar otros descriptores (filtros) diferentes al proceso evolutivo para diseñar nuevos descriptores fenotípicos que pudieran superar el desempeño de los obtenidos hasta este momento. Igualmente, esta metodología se podrá aplicar a otras bases de datos para reconocimiento de emociones.

Referencias

- [1] Amal Adouani, Wiem Mimoun Ben Henia y Zied Lachiri. “Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences”. En: *2019 16th International Multi-Conference on Systems, Signals & Devices (SSD)*. IEEE, 2019, págs. 266-271.
- [2] Taha J Alhindi y col. “Comparing LBP, HOG and deep features for classification of histopathology images”. En: (mayo de 2018). arXiv: 1805.05837 [eess.IV].
- [3] Malika Arora, Munish Kumar y Naresh Kumar Garg. “Facial emotion recognition system based on PCA and gradient features”. en. En: *Natl. Acad. Sci. Lett.* 41.6 (dic. de 2018), págs. 365-368.
- [4] Hadjer Boubenna y Dohoon Lee. “Image-based emotion recognition using evolutionary algorithms”. En: *Biol. Inspired Cogn. Arch.* 24 (abr. de 2018), págs. 70-76.
- [5] Valentin Calzada-Ledesma y col. “Evolutionary design of problem-adapted image descriptors for texture classification”. En: *IEEE Access* 6 (2018), págs. 40450-40462.
- [6] J Y R Cornejo y H Pedrini. “Emotion recognition from occluded facial expressions using weber local descriptor”. En: *25Th International Conference on Systems, Signals and Image Processing*. IEEE, 2018, págs. 1-5.
- [7] J De Anda-Suárez y col. “Symmetric-approximation Energy-based Estimation of Distribution (SEED): a continuous optimization algorithm”. En: *IEEE Access* 7 (2019), págs. 154859-154871.
- [8] P E Ekman y R J Davidson. *The nature of emotion: Fundamental questions*. Oxford University Press, 1994, págs. 20-31.

- [9] S K Eng y col. “Facial expression recognition in JAFFE and KDEF Datasets using histogram of oriented gradients and support vector machine”. En: *IOP Conf. Ser. Mater. Sci. Eng.* 705.1 (nov. de 2019), págs. 012031.
- [10] David Fagan y Eoin Murphy. “Mapping in grammatical evolution”. En: *Handbook of Grammatical Evolution*. Springer, 2018, págs. 79-108.
- [11] Yin Fan y col. “Video-based emotion recognition using CNN-RNN and C3D hybrid networks”. En: *Proceedings of the 18th ACM International Conference on Multimodal Interaction - ICMI 2016*. Tokyo, Japan: ACM Press, 2016.
- [12] V Franzoni y A Milani. “Emotion recognition for self-aid in addiction treatment, psy- chotherapy, and nonviolent communication”. En: *International Conference on Computational Science and Its Applications*. Cham: Springer, 2019, págs. 391-404.
- [13] Benyamin Ghojogh y col. “Reproducing Kernel Hilbert Space, Mercer’s Theorem, Eigenfunctions, Nystrom Method, and Use of Kernels in Machine Learning: Tutorial and Survey”. En: *arXiv preprint arXiv:2106.08443* (2021).
- [14] Hadeel Tariq Ibrahim y col. “A grasshopper optimizer approach for feature selection and optimizing SVM parameters utilizing real biomedical data sets”. En: *Neural Computing and Applications* 31.10 (2019), págs. 5965-5974.
- [15] Deepak Kumar Jain, Pourya Shamsolmoali y Paramjit Sehdev. “Extended deep neural network for facial emotion recognition”. En: *Pattern Recognition Letters* 120 (2019), págs. 69-74.
- [16] V Jain y col. “Facial expression analysis and the affect space”. en. En: *Pattern Recognit. Image Anal.* 25.3 (jul. de 2015), págs. 430-436.

- [17] J Kulandai Josephine Julina y T Sree Sharmila. “Facial Emotion Recognition in Videos using HOG and LBP”. En: *2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*. IEEE, 2019, págs. 56-60.
- [18] Durga Ganga Rao Kola y Srinivas Kumar Samayamantula. “A novel approach for facial expression recognition using local binary pattern with adaptive window”. en. En: *Multimed. Tools Appl.* 80.2 (ene. de 2021), págs. 2243-2262.
- [19] D Lakshmi y R Ponnusamy. “Facial emotion recognition using modified HOG and LBP features with deep stacked autoencoders”. En: *Microprocessors and Microsystems* 82 (2021), pág. 103834.
- [20] Roisin Loughran y Michael O’Neill. “Evolutionary music: applying evolutionary computation to the art of creating music”. En: *Genetic Programming and Evolvable Machines* 21.1 (2020), págs. 55-85.
- [21] P Lucey y col. “The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression”. En: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE, 2010, págs. 94-101.
- [22] D Lundqvist, A Flykt y A Ohman. “CD ROM from Department of Clinical Neuroscience, Psychology section”. En: *Karolinska Institutet* 91.630 (1998), págs. 2-2.
- [23] M Lyons, M Kamachi y J Gyoba. *Japanese female facial expression (JAFFE) database*. 2017.
- [24] K B Manas. *Computer Vision and Image Processing*. Taylor Francis Group, 2019, págs. 215-217.

- [25] K B Manas. *Computer Vision and Image Processing*. Taylor Francis Group, 2019.
- [26] Gonzalo Pajares Martinsanz. *Visión por computador. Imágenes Digitales y Aplicaciones*. Alfaomega, 2008.
- [27] Ryan Melaugh y col. “Facial Expression Recognition on partial facial sections”. En: *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*. Dubrovnik, Croatia: IEEE, sep. de 2019.
- [28] K Mohan y col. “Facial expression recognition using local gravitational force descriptor-based deep convolution neural networks”. En: *IEEE Transactions on Instrumentation and Measurement* 70 (2020), págs. 1-12.
- [29] Jason H Moore y Moshe Sipper. “Grammatical evolution strategies for bioinformatics and systems genomics”. En: *Handbook of Grammatical Evolution*. Cham: Springer International Publishing, 2018, págs. 395-405.
- [30] Seetha Parameswaran y col. “Exploring various aspects of Gabor filter in classifying facial expression”. En: *Lecture Notes in Electrical Engineering*. Lecture notes in electrical engineering. Singapore: Springer Singapore, 2020, págs. 487-500.
- [31] D A Pitaloka y col. “Enhancing CNN with preprocessing stage in automatic emotion recognition”. En: *Procedia computer science* 116 (2017), págs. 523-529.
- [32] O Quiroz-Ramírez y col. “Partially-connected artificial neural networks developed by grammatical evolution for pattern recognition problems. In: Fuzzy Logic Augmentation of Neural and Optimization Algorithms: Theoretical Aspects and Real Applications”. En: *Studies in Computational Intelligence* 749 (2013), págs. 293-305.

- [33] T S Saj y col. “Facial Emotion Recognition Using Shallow CNN”. En: *Symposium on Machine Learning and Metaheuristics Algorithms, and Applications*. Singapore: Springer, 2019, págs. 144-150.
- [34] Mamta Santosh y Avinash Sharma. “A proposed framework for emotion recognition using Canberra distance classifier”. En: *Journal of Computational and Theoretical Nanoscience* 16.9 (2019), págs. 3778-3782.
- [35] Dan Simon. *Evolutionary optimization algorithms*. Nashville, TN: John Wiley & Sons, jun. de 2014.
- [36] Teck Yan Tan y col. “Evolving ensemble models for image segmentation using enhanced particle swarm optimization”. En: *IEEE Access* 7 (2019), págs. 34004-34019.
- [37] Paweł Tarnowski y col. “Emotion recognition using facial expressions”. en. En: *Procedia Comput. Sci.* 108 (2017), págs. 1175-1184.
- [38] Cigdem Turan y Kin-Man Lam. “Histogram-based local descriptors for facial expression recognition (FER): A comprehensive study”. en. En: *J. Vis. Commun. Image Represent.* 55 (ago. de 2018), págs. 331-341.
- [39] Fei Wang y col. “Comparative study on KNN and SVM based weather classification models for day ahead short term solar PV power forecasting”. En: *Appl. Sci. (Basel)* 8.1 (dic. de 2017), pág. 28.
- [40] Phillip Ian Wilson y John Fernandez. “Facial feature detection using Haar classifiers”. En: *Journal of Computing Sciences in Colleges* 21.4 (2006), págs. 127-133.

- [41] Zheng Xiang, Hengliang Tan y Wenling Ye. “The excellent properties of a dense grid-based HOG feature on face recognition compared to Gabor and LBP”. En: *IEEE Access* 6 (2018), págs. 29306-29319.
- [42] Dongri Yang y col. “An emotion recognition model based on facial recognition in virtual learning environment”. En: *Procedia Computer Science* 125 (2018), págs. 2-10.
- [43] Jucheng Yang y col. “Face expression recognition using Gabor features and a novel Weber local descriptor”. En: *Biometric Recognition. Lecture notes in computer science*. Cham: Springer International Publishing, 2018, págs. 265-274.
- [44] Thein Yu y Khin Thandar Nwet. “Comparing SVM and KNN algorithms for Myanmar news sentiment analysis system”. En: *Proceedings of 2020 the 6th International Conference on Computing and Data Engineering*. 2020, págs. 65-69.

7. Anexo A

Diseño de descriptores mediante Evolución Gramatical para el reconocimiento de imágenes de expresiones faciales.

Diseño de descriptores mediante Evolución Gramatical para el reconocimiento de imágenes de expresiones faciales.

Resumen. El reconocimiento de expresiones faciales representa una importante tarea en las áreas de visión por computadora y reconocimiento de patrones. La obtención de características suficientemente discriminativas para reconocer diferentes emociones resulta ser una etapa de suma dificultad. En este trabajo se presenta una metodología basada en Evolución Gramatical e Histograma de Gradientes Orientados para la generación automática de descriptores capaces de caracterizar adecuadamente imágenes de expresiones faciales.

Palabras clave: Reconocimiento de expresiones faciales, descriptores de imágenes, Evolución Gramatical, Histograma de Gradientes Orientados.

Descriptors Design using Grammatical Evolution for Facial Expression images Recognition.

Abstract: Recognition of facial expressions represents an important task in areas of computer vision and pattern recognition. Obtaining sufficiently discriminative characteristics to recognize different emotions turns out to be a very difficult stage. This work presents a methodology based on Grammatical Evolution and Histogram of Oriented Gradients for the automatic generation of descriptors capable of properly characterizing images of facial expressions.

Keywords: Face emotion recognition, Images descriptors, Grammatical Evolution, Histogram of oriented gradients.

1. Introducción

Dentro de la comunicación humana, las expresiones faciales toman un rol muy importante al momento de entablar una conversación; un gran porcentaje del mensaje que se quiere comunicar se muestra a través de estas expresiones, por lo que se da un indicio del contexto de la información que se está transmitiendo, así como también del estado de ánimo de la persona.

Actualmente, el estudio formal de las emociones se ha incrementado y existen una gran cantidad de autores que han contribuido en esta área de estudio. Uno de los aportes más importantes fue realizado por Ekman, en su libro titulado: “*The nature of emotion: Fundamental questions*”, en donde se propuso una estandarización de las emociones, formando un grupo de seis emociones básicas: enojo, miedo, disgusto, felicidad, tristeza y sorpresa [1]. La estandarización de las emociones humanas fue uno de los puntos de partida que hicieron posible el reconocimiento de expresiones faciales. Esto permitió un nuevo enfoque para la interacción humano máquina. De esta manera, ha sido posible adaptar herramientas de visión por computadora para el análisis de las emociones, lo que ha permitido el desarrollo de diversos instrumentos que ayudan, por ejemplo, a la identificación de un posible agresor, la predicción del comportamiento de personas en hospitales o instituciones psiquiátricas, sistemas de reconocimiento biométrico, sistemas de seguridad en aeropuertos, etc [2].

Cualquier sistema de reconocimiento de expresiones faciales usualmente se basa en las siguientes fases (ver Fig. 1): Adquisición de imágenes de rostros, Extracción de características y Clasificación.

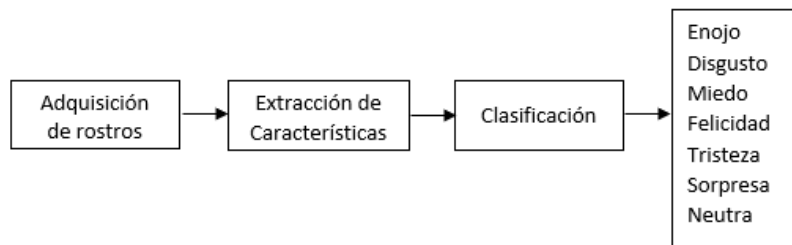


Fig. 1. Sistema de reconocimiento de expresiones faciales[3].

La primera fase consiste en la obtención y preprocesamiento de las imágenes de los rostros o secuencias de videos, en el preprocesamiento se realiza un mejoramiento de las imágenes adquiridas previo a la extracción de características, por ejemplo se puede segmentar y eliminar el fondo de las imágenes para únicamente tomar en cuenta zonas de interés en la imagen. Otro ejemplo es la aplicación de filtros que permitan eliminar el ruido causado por la iluminación.

La extracción de características es una de las fases más importantes dentro del sistema, usualmente se lleva a cabo a través de la aplicación de descriptores los cuales son algoritmos computacionales que extraen y empaquetan los atributos significativos de las imágenes en vectores de características. Es deseable que estos atributos caracterizados permitan una buena separabilidad entre clases (emociones), permitiendo al clasificador tener un buen desempeño. Finalmente, en la fase de clasificación, los vectores de características se etiquetan o asocian a una clase cuyas características son similares; esto se realiza mediante los algoritmos clasificadores, los cuales utilizan un indicador (porcentaje de clasificación) que permite conocer la cantidad de vectores que fueron asociados con sus respectivas clases de manera correcta.

Existe una variedad de descriptores que pueden ser utilizados para caracterizar imágenes, por ejemplo, Local Binary Pattern (LBP), Filtros de Gabor, Histograma de Gradientes Orientados (HOG por sus siglas en inglés), entre otros. Estos algoritmos han sido diseñados por expertos en el campo para abordar un determinado tipo de problemas, sin embargo, es común observar que el rendimiento de éstos puede ser menor cuando son aplicados a otros diferentes tipos de problemas; de esta manera se puede decir que no existe un descriptor perfecto. Este hecho ha motivado a la creación de metodologías para la generación automática de descriptores que se adapten a un conjunto de imágenes en particular para poder obtener un mejor desempeño en el proceso de clasificación [4].

En este artículo se propone una metodología basada en Evolución Gramatical y HOG, para la generación automática de descriptores aplicados al reconocimiento de expresiones faciales, utilizando como instancia de prueba una de las bases de datos más citadas en el estado del arte, el Japanese Female Facial Expression (JAFPE). Una Máquina de Vector Soporte es utilizada como algoritmo de clasificación.

El trabajo está organizado de la siguiente manera: la sección 2 muestra conceptos teóricos relacionados con la presente propuesta; la sección 3 describe la metodología empleada; en la sección 4 se explican los experimentos realizados y los resultados obtenidos y finalmente en la sección 5 se presentan las conclusiones.

2. Marco Teórico

Evolución Gramatical. Evolución Gramatical (GE por sus siglas en inglés) es considerada una variación de la Programación Genética [10], la cual utiliza una gramática formal relacionada con el problema para generar un fenotipo o programa, el cual es codificado a partir de un arreglo numérico denominado cromosoma genotípico [10]. Además, usa un proceso de mapeo, el cual permite relacionar los elementos de la gramática con cada cromosoma para obtener un fenotipo. Un motor de búsqueda realiza el proceso evolutivo.

Gramática libre de contexto tipo Backus-Naur. La Gramática tipo Backus-Naur (BNF) es una notación formal que permite diseñar la sintaxis de un programa basándose en un conjunto de reglas de producción [10], estas reglas permiten la obtención del programa (fenotipo) tomando en cuenta símbolos divididos en terminales y no terminales. Los elementos que conforman la gramática son expresados mediante la tupla $\{N, T, P, S\}$, donde N representa los símbolos no terminales (indicados en la gramática entre signos ' \langle ' y ' \rangle '), T indica los símbolos terminales, P es el conjunto de reglas de producción y S es el símbolo inicial que indica la primera regla de producción que se aplica [11].

Proceso de Mapeo. El proceso de mapeo se encarga de convertir un cromosoma genotípico en un fenotipo basándose en los elementos que conforman la gramática BNF (N, T, P, S); cada elemento del cromosoma recibe el nombre de codón. El algoritmo Depth-First (DF) fue usado como proceso de mapeo, el cual utiliza una regla de derivación que recorre los símbolos no terminales de la gramática hasta encontrar símbolos terminales. En la Fig. 2 se puede observar la regla de derivación y

el proceso de mapeo, el cual recibe un codón para aplicar la regla a la gramática e ir formando el árbol de derivación hasta obtener el fenotipo.

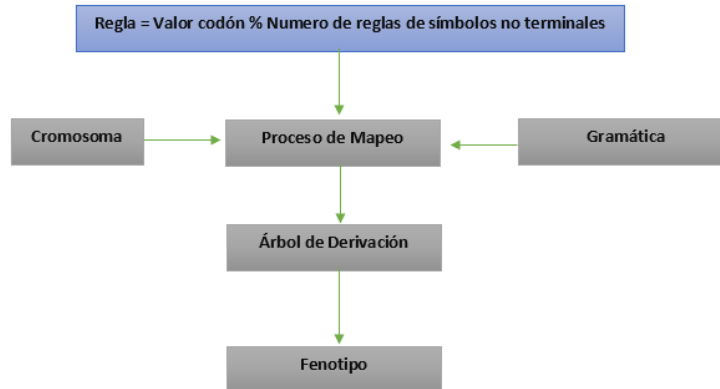


Fig. 2. Diagrama del proceso de mapeo [11].

A continuación, se muestra un ejemplo de este proceso de mapeo, utilizando la gramática BNF de la Fig. 3, la cual se encarga de la creación de operaciones entre números complejos. En la gramática el símbolo inicial S es: $\langle NumComplejo \rangle$, los terminales T son: $i, +, -, *, /, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0$, los símbolos no terminales N son: $\langle oper \rangle, \langle op \rangle, \langle signo \rangle, \langle OpReal \rangle, \langle numero \rangle$ y el conjunto de reglas de producción P se muestra en la Fig. 3.

```

<NumComplejo> ::= (<oper><op><oper>) (0)
<oper> ::= <signo><OpReal><signo><OpReal> i (0)
<signo> ::= + | (0)
           - | (1)
<OpReal> ::= <OpReal><numero> | (0)
           <numero> (1)
<numero> ::= 1 | (0)
           2 | (1)
           3 | (2)
           4 | (3)
           5 | (4)
           6 | (5)
           7 | (6)
           8 | (7)
           9 | (8)
           0 | (9)
<op> ::= + | (0)
        - | (1)
        * | (2)
        / | (3)
  
```

Fig. 3. Ejemplo de Gramática BNF.

El proceso de mapeo DF comienza con el símbolo inicial y procede a sustituir el símbolo no terminal localizado más a la izquierda, usando la regla de derivación basada en el operador módulo, hasta que encuentra un símbolo terminal. Enseguida se desplaza hacia la derecha para sustituir el siguiente no terminal. Este proceso continúa hasta que todos los símbolos no terminales del fenotipo generado son reemplazados por símbolos terminales.

En el ejemplo mostrado en la Fig. 4, el primer codón del cromosoma es 15 y el símbolo no terminal inicial tiene solo una regla de producción; aplicando la regla de derivación se obtiene $15\%1=0$, por lo que se selecciona la producción 0: $(\langle oper \rangle)\langle op \rangle(\langle oper \rangle)$. El siguiente codón es 3 y el símbolo no terminal más a la izquierda $(\langle oper \rangle)$ tiene también una producción válida, dando como resultado $3\%1=0$, seleccionando $\langle signo \rangle \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i$. Como se mencionó anteriormente, el proceso de mapeo finaliza cuando ya no existen símbolos no terminales en el fenotipo.

Cromosoma	15	3	31	25	61	48	7	5	12	16	3	17	36	22	41	18	
$(\langle oper \rangle)\langle op \rangle(\langle oper \rangle)$																	$15\%1=0$
$(\langle signo \rangle \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i)\langle op \rangle(\langle oper \rangle)$																	$3\%1=0$
$(-\langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i)\langle op \rangle(\langle oper \rangle)$																	$31\%2=1$
$(-\langle numero \rangle \langle signo \rangle \langle OpReal \rangle i)\langle op \rangle(\langle oper \rangle)$																	$25\%2=1$
$(-2 \langle signo \rangle \langle OpReal \rangle i)\langle op \rangle(\langle oper \rangle)$																	$61\%10=1$
$(-2 + \langle OpReal \rangle i)\langle op \rangle(\langle oper \rangle)$																	$48\%2=0$
$(-2 + \langle numero \rangle i)\langle op \rangle(\langle oper \rangle)$																	$7\%2=1$
$(-2 + 6 i)\langle op \rangle(\langle oper \rangle)$																	$5\%10=5$
$(-2 + 6 i) + (\langle oper \rangle)$																	$12\%4=0$
$(-2 + 6 i) + (\langle signo \rangle \langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i)$																	$16\%1=0$
$(-2 + 6 i) + (-\langle OpReal \rangle \langle signo \rangle \langle OpReal \rangle i)$																	$3\%2=1$
$(-2 + 6 i) + (-\langle numero \rangle \langle signo \rangle \langle OpReal \rangle i)$																	$17\%2=1$
$(-2 + 6 i) + (-7 \langle signo \rangle \langle OpReal \rangle i)$																	$36\%10=6$
$(-2 + 6 i) + (-7 + \langle OpReal \rangle i)$																	$22\%2=0$
$(-2 + 6 i) + (-7 + \langle numero \rangle i)$																	$41\%2=1$
$(-2 + 6 i) + (-7 + 9 i)$																	$18\%10=8$

Fig. 4. Proceso de mapeo DF para obtener operaciones de números complejos.

Motor de Búsqueda. El motor de búsqueda consiste en un algoritmo de optimización capaz de encontrar soluciones adecuadas dentro de un determinado espacio de búsqueda. En este trabajo se empleó el algoritmo de Evolución Diferencial (DE por sus siglas en inglés) como motor de búsqueda. Para obtener una solución óptima, DE evoluciona un conjunto de soluciones candidatas a ser la solución mejor adaptada al problema. A cada solución se les conoce como individuo y al conjunto de individuos se les denomina población [12]. En la Fig. 5 se muestra el algoritmo de DE.

```

F=coeficiente de mutación ∈ [0.4,0.9]
C=coeficiente de cruza ∈ [0.1,1]
Inicializar población de soluciones candidatas {xi} para i desde 1 hasta N
Repetir mientras sea diferente(criterio de paro)
  Repetir por cada individuo xi para i desde 1 hasta N
    r1 ← entero aleatorio ∈ [1,N] : r1 ≠ i
    r2 ← entero aleatorio ∈ [1,N] : r2 ≠ {i, r1}
    r3 ← entero aleatorio ∈ [1,N] : r3 ≠ {i, r1, r3}
    vi ← xr1 + F(xr2 - xr3) vector mutado
    jr ← entero aleatorio ∈ [1, n]
    Repetir por cada dimensión para j desde 1 hasta n
      rcj ← numero aleatorio ∈ [0,1]
      Si (rcj < C) o (j == jr)
        vij ← vij
      De otro modo
        vij ← xij
      Termina condicional "Si"
    Siguiete dimensión
  Siguiete individuo
  Repetir por cada población desde i igual a 1 hasta N
  Si f(vi) < f(xi)
    xi ← vi
  Siguiete población
  Siguiete generación

```

Fig. 5. Pseudocódigo de Evolución Diferencial [12].

Histograma de Gradientes Orientados. Histograma de Gradientes Orientados (HOG) es un descriptor que utiliza la magnitud y la orientación de una imagen sobre los pixeles de una pequeña región denominada “celda”; estas cantidades son concatenadas construyendo un histograma de una dimensión tomando en cuenta los niveles de gris de los pixeles y usando la ecuación (1) [5]:

$$\theta_{x,y} = \tan^{-1} \frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)} \quad (1)$$

Donde: $L = \text{nivel de intensidad del pixel en la posición de la celda}$

Las orientaciones son acumuladas en un número N de bins (cantidad de elementos a tomar en cuenta por histograma) que nos sirve para posteriormente crear el histograma (ver Fig. 6); el vector de características se forma al concatenar los histogramas creados.

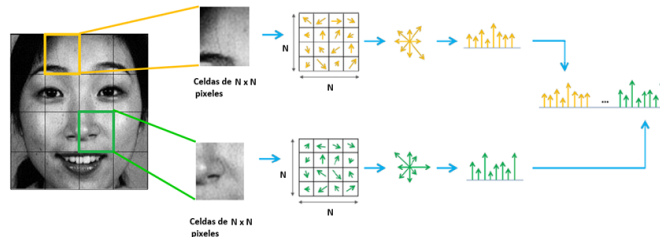


Fig. 6. Caracterización de una imagen con HOG.

3. Metodología

En esta sección se explica el método propuesto basado en GE para la caracterización automática de imágenes de expresiones faciales. En la Fig. 7 se muestra el diagrama de dicha metodología.

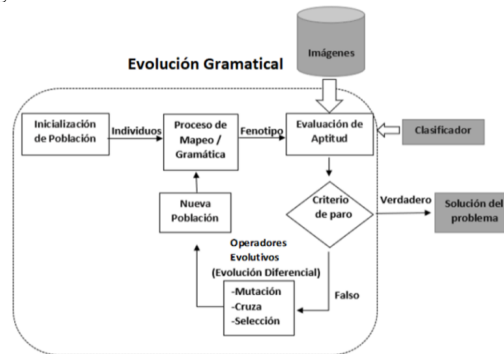


Fig. 7. Diagrama de la metodología propuesta [4].

Las entradas requeridas por GE son: el conjunto de imágenes de la base de datos JAFFE y un algoritmo clasificador (en este trabajo se empleó una Máquina de Vector Soporte). Inicialmente se crea una población de individuos (cromosomas) con una determinada dimensión y con valores numéricos enteros aleatorios. Por medio de la gramática BNF y el proceso de mapeo Depth-First, los individuos se transforman a su forma fenotípica y su calidad es evaluada por una función de aptitud. Luego, los operadores de mutación, cruza y selección pertenecientes a DE son aplicados a la población para obtener nuevos individuos que puedan mejorar su desempeño en el problema. Este proceso se repite hasta que algún criterio de paro se cumpla (en nuestro caso se utilizó un determinado número de generaciones).

Gramática BNF. Los descriptores generados por GE se forman por la combinación de un conjunto de operadores básicos agrupados en filtros pasa-alta, pasa-baja, direccionales, Max-Pooling, escala de grises y operadores aritméticos [4]. Estos operadores constituyen los símbolos terminales de la gramática BNF. En la Tabla 1 se muestra una descripción de estos operadores y sus correspondientes símbolos empleados en la gramática.

Tabla 1. Operadores básicos utilizados de la Gramática BNF.

Tipo de operador	Nombre	Símbolo terminal en la Gramática
Pasa-alta	Laplaciano	<i>Lap</i>
Pasa-alta	Laplaciano de la Gaussiana	<i>LapG1, LapG2</i>
Pasa-alta	Derivada de la Gaussiana	<i>GauDX, GauDY</i>
Pasa-baja	Gaussiana	<i>Gau1, Gau2</i>

Pasa-baja	Media	<i>AverF</i>
Pasa-baja	Mediana	<i>MedianF</i>
Direccional	Banco de Filtros de Gabor	<i>GFB0, GFB45, GFB90, GFB135</i>
Max-Pooling	Max Pooling	<i>MP2, MP4, MP6, MP8, MP10</i>
	Ecuación del Histograma	<i>HEq</i>
Operador aritmético	Valor Absoluto de una imagen	<i>AbsV</i>
Operador aritmético	Cuadrado de una imagen	<i>Sqr</i>
Operador aritmético	Raíz cuadrada de una imagen	<i>Sqrt</i>
Operador aritmético	Logaritmo base 2	<i>Log</i>
Operador aritmético	Multiplicar una imagen por 0.5	<i>T0.5</i>
Operador aritmético	Sumar 2 imágenes	<i>ADD</i>
Operador aritmético	Restar 2 imágenes	<i>SUB</i>
Operador aritmético	Resta absoluta de 2 imágenes	<i>ASUB</i>
Operador aritmético	Multiplicar 2 imágenes	<i>MUL</i>
Operador aritmético	Dividir 2 imágenes	<i>DIV</i>
Escala de grises	Obtener imagen a escala de grises	<i>lg</i>

Los símbolos no terminales empleados en la gramática BNF fueron los siguientes:

$N = \{ \langle Start \rangle, \langle MP \rangle, \langle Expr \rangle, \langle Filter \rangle, \langle Gau \rangle, \langle Lap \rangle, \langle GFB \rangle, \langle Arith \rangle, \langle Op \rangle, \langle Terminal \rangle \}$

Las reglas de producción se muestran en la Tabla 2, las cuales son separadas por el símbolo “|”. Por lo tanto, un símbolo no terminal puede tener una o varias reglas de producción, las cuales se enumeran para el proceso de mapeo. El símbolo inicial es $\langle Start \rangle$.

Tabla 2. Reglas de producción utilizadas en la Gramática BNF [4].

Símbolo no terminal	Reglas de producción
$\langle Start \rangle$	$\langle Expr \rangle \langle MP \rangle$ (0)
$\langle Expr \rangle$	$\langle Expr \rangle \langle Expr \rangle \langle Op \rangle$ (0) $\langle Expr \rangle \langle Filter \rangle$ (1) $\langle Terminal \rangle$ (2)
$\langle Op \rangle$	ADD (0) SUB (1) $ASUB$ (2) MUL (3) DIV (4)
$\langle Filter \rangle$	$\langle Gau \rangle$ (0) $\langle Lap \rangle$ (1) $\langle GFB \rangle$ (2) $\langle Arith \rangle$ (3)
$\langle Gau \rangle$	$Gau1$ (0) $Gau2$ (1) $GauDX$ (2)

	<i>GauDY</i> (3)
<Lap>	<i>LapG1</i> (0) <i>LapG2</i> (1) <i>Lap</i> (2)
<GFB>	<i>GFB0</i> (0) <i>GFB45</i> (1) <i>GFB90</i> (2) <i>GFB135</i> (3)
<Arith>	<i>AverF</i> (0) <i>MedianF</i> (1) <i>HEq</i> (2) <i>AbsV</i> (3) <i>Sqr</i> (4) <i>Sqrt</i> (5) <i>Log</i> (6) <i>T0.5</i> (7)
<MP>	<i>MP2</i> (0) <i>MP4</i> (1) <i>MP6</i> (2) <i>MP8</i> (3) <i>MP10</i> (4)
<Terminal>	<i>lg</i> (0)

Función de aptitud. Una vez que se tiene un fenotipo (descriptor generado por GE), se requiere evaluar su calidad mediante una función de aptitud. Para ello se aplica el descriptor a la base de datos JAFFE y se obtiene un nuevo conjunto de imágenes procesadas. Luego, para cada imagen procesada se obtiene un vector de características mediante HOG, el cual es etiquetado de acuerdo a la clase a la que pertenece. Finalmente el conjunto de vectores es clasificado por una Máquina de Vector Soporte (SVM) y el porcentaje de clasificación obtenido representa la aptitud del fenotipo.

Un ejemplo de la aplicación de un descriptor generado por GE a una imagen de la base de datos JAFFE se muestra en la Fig. 8. El descriptor generado es “*lg*, *GauDY*, *lg*, *SUB*, *Gau2*, *MP6*” y mediante notación postfija se va aplicando paso por paso y de izquierda a derecha a la imagen de prueba. A la imagen final procesada se le aplica HOG para obtener un vector de características.

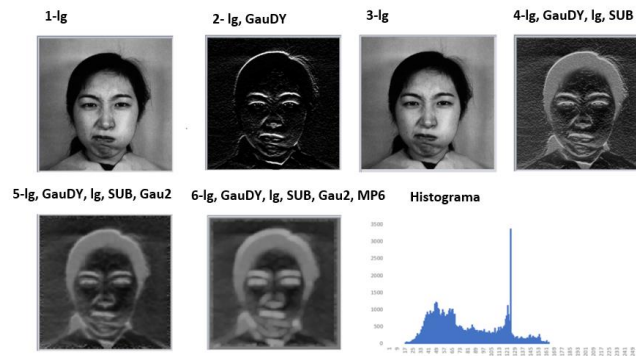


Fig. 8. Ejemplo del proceso de caracterización de una imagen.

4. Experimentación y resultados.

En esta sección se muestra la experimentación realizada y los resultados obtenidos de la misma.

Base de datos. Japanese Female Facial Expressions (JAFFE) fue la base de datos utilizada en este trabajo [9]. Contiene 213 imágenes de rostros de 10 mujeres

japonesas, agrupadas en 6 expresiones faciales básicas además de una expresión neutra, quedando distribuidas como sigue: 30 imágenes neutrales, 30 imágenes de enojo, 29 imágenes de disgusto, 32 imágenes de miedo, 31 imágenes de alegría, 31 imágenes de tristeza y 30 imágenes de sorpresa. Cada imagen tiene una dimensión de 256×256 píxeles.

Experimentación. Para probar la metodología propuesta y tener consistencia en los resultados obtenidos, se realizaron 35 experimentos de manera independiente. A continuación, se muestran los valores de los parámetros utilizados en los diferentes algoritmos involucrados en la generación automática de descriptores, así como del algoritmo clasificador.

Población de individuos: El tamaño de la población fue de 20 cromosomas y cada uno de ellos estuvo formado por 20 códons con valores enteros en el rango de [0, 255].

Motor de búsqueda ED: Coeficiente de mutación $F=0.5$ y coeficiente de cruce $C=0.8$ [4].

Número de generaciones del proceso evolutivo = 80

Algoritmo HOG: Tamaño de celdas = 32×32 píxeles y tamaño de histogramas = 5 bins.

Algoritmo clasificador (SVM): Kernel polinomial, tolerancia 0.001. El algoritmo SVM fue obtenido del software Waikato Environment for Knowledge Analysis (WEKA).

En el proceso de clasificación se utilizó validación cruzada con $K = 10$ carpetas para obtener un porcentaje de clasificación promedio como valor de aptitud del descriptor generado por GE.

Resultados. El porcentaje de clasificación de nuestra propuesta se presenta como el promedio de los mejor resultados de aptitud de los 35 experimentos realizados en forma independiente y se compara en la Tabla 3 con resultados del estado de arte.

Tabla 3. Comparación de resultados del dataset JAFFE.

Autor	Metodología	Porcentajes de Clasificación
Eng. Et. al.[5]	HOG , SVM	88.89 %
Yang Et. al. [6]	Banco de Filtros de Gabor + Descriptor Local de Weber , KNN	92.77 %
Pitaloka Et. al. [7]	Deep Learning	97.06 %
Melaugh Et. al. [8]	Red Neuronal Convolucional	76.56 %
Nuestra propuesta	Evolución Gramatical + HOG , SVM	97.78 %

En esta tabla se puede observar que nuestra metodología usando GE obtuvo resultados semejantes a la de Deep Learning [7]. La realización de un análisis estadísticos para comparar el rendimiento de esta propuesta con respecto a las del

estado de arte no fue factible debido a que cada metodología propone experimentos de manera distinta.

Finalmente, en la Fig. 9 se muestra la evolución del mejor valor de aptitud en función del número de generación para un experimento, utilizando los valores de parámetros reportados al inicio de esta sección.

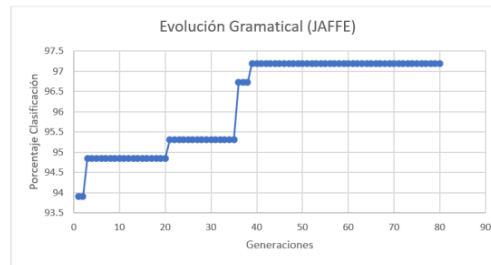


Fig. 9. Gráfica de evolución del mejor valor de aptitud por generación.

5. Conclusiones

En este artículo se propuso una metodología basada en Evolución Gramatical e Histograma de Gradientes Orientados para la generación automática de descriptores aplicados al reconocimiento de imágenes de expresiones faciales utilizando la base de datos JAFPE. Evolución Gramatical utilizó una gramática tipo Backus-Naur, Evolución Diferencial como motor de búsqueda y el proceso de mapeo Depth-First para generar descriptores sintácticamente correctos que se aplicaron a las imágenes de la base de datos. Posteriormente las imágenes procesadas fueron caracterizadas utilizando el Histograma de Gradientes Orientados y los vectores obtenidos se clasificaron mediante una Máquina de Vector Soporte.

Los resultados obtenidos mostraron que, para el caso de la base de datos JAFPE, la presente propuesta tuvo un desempeño comparable con metodologías del estado de arte e incluso superó los resultados de algunas de ellas. Adicionalmente, esta metodología no requiere conocimiento a priori del problema a resolver ni intervención de un experto.

Como trabajo a futuro, se desea aplicar la metodología a otras bases de datos del estado del arte, como es el caso de Extended Cohn-Kanade (CK+) o Karolinska Directed Emotional Faces (KDEF). Además, se realizará un análisis comparativo de diferentes algoritmos metaheurísticos que puedan ser usados como motor de búsqueda en Evolución Gramatical.

Referencias

1. Ekman, P. E., & Davidson, R. J.: *The nature of emotion: Fundamental questions*, Oxford University Press, pp. 20-31, (1994).

2. Cornejo, J. Y. R., & Pedrini, H.: Emotion recognition from occluded facial expressions using weber local descriptor. In: 25Th international conference on systems, signals and image processing (IWSSIP), IEEE, pp. 1-5, (2018)
3. Kola, D. G. R., & Samayamantula, S. K.: A novel approach for facial expression recognition using local binary pattern with adaptive window, *Multimedia Tools and Applications*, pp. 1-20, (2020).
4. Calzada-Ledesma, V., Puga-Soberanes, H. J., Ornelas-Rodriguez, M., Rojas-Dominguez, A., Carpio-Valadez, J. M., Espinal, A., ... & Sotelo-Figueroa, M. A.: Evolutionary design of problem-adapted image descriptors for texture classification, *IEEE Access*, vol 6, pp. 40450-40462, (2018).
5. Eng, S. K., Ali, H., Cheah, A. Y., & Chong, Y. F.: Facial expression recognition in JAFFE and KDEF datasets using histogram of oriented gradients and support vector machine. In: *IOP Conference series: materials science and engineering*, IOP Publishing, Vol. 705, p. 012031, (2019)
6. Yang, J., Li, M., Zhang, L., Han, S., Wang, X., & Wang, J.: Face Expression Recognition Using Gabor Features and a Novel Weber Local Descriptor. In: *Chinese Conference on Biometric Recognition*, Springer, Cham, pp. 265-274, p. 012031: (2018).
7. Pitaloka, D. A., Wulandari, A., Basaruddin, T., & Liliana, D. Y.: Enhancing CNN with preprocessing stage in automatic emotion recognition, *Procedia computer science*, vol. 116, pp. 523-529, (2017).
8. Melaugh, R., Siddique, N., Coleman, S., & Yogarajah, P.: Facial expression recognition on partial facial sections. In: *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, IEEE, pp. 193-197, (2019).
9. M. Lyons, M. Kamachi, and J. Gyoba : “Japanese Female Facial Expressions (JAFFE),” Database of Digital Images, <https://zenodo.org/record/3451524#.YHCK3-gzZPY>, (1997).
10. Moore, J. H., & Sipper, M.: Grammatical Evolution Strategies for Bioinformatics and Systems Genomics. In *Handbook of Grammatical Evolution*, Springer, pp. 395-405, (2018).
11. Quiroz-Ramírez, O., Espinal, A., Ornelas-Rodríguez, M., Rojas-Domínguez, A., Sánchez, D., Puga-Soberanes, H., Carpio, M., Mancilla Espinoza, L. & Ortíz-López, J.: Partially-connected artificial neural networks developed by grammatical evolution for pattern recognition problems. In: *Fuzzy Logic Augmentation of Neural and Optimization Algorithms: Theoretical Aspects and Real Applications*, Studies in Computational Intelligence 749, Springer, pp. 99-112, (2018).
12. Simon, D.: *Evolutionary optimization algorithms*. John Wiley & Sons, pp. 293-305, (2013)



LA SOCIEDAD MEXICANA DE INTELIGENCIA ARTIFICIAL
Y LA UNIVERSIDAD DE SONORA

OTORGAN ESTE CERTIFICADO A:

Manuel Alejandro Torres Fonseca, Valentín Calzada Ledesma, Manuel Ornelas Rodríguez, Alfonso Rojas Domínguez, Héctor José Puga Soberanes y Juan Martín Carpio Valadez

por la presentación del artículo titulado:

Diseño de descriptores mediante Evolución Gramatical para el reconocimiento de imágenes de expresiones faciales

en el XIII Congreso Mexicano de Inteligencia Artificial, COMIA 2021

Hermosillo, Sonora, México, del 18 al 21 de mayo de 2021

Dr. Felix Castro Espinoza
Presidente SMIA

Dr. Oscar Herrera Alcántara
Presidente del Comité de Programa

Dr. Noé A. Castro Sánchez
Presidente del Comité de Programa

Dra. Maria Elena Robles Baldenegro
Comité Local COMIA