

Diseño de descriptores mediante evolución gramatical para el reconocimiento de imágenes de expresiones faciales

Manuel Alejandro Torres Fonseca¹, Valentín Calzada Ledesma²,
Manuel Ornelas Rodríguez¹, Alfonso Rojas Domínguez¹,
Juan Martín Carpio Valadez¹, Héctor José Puga Soberanes¹

¹ Tecnológico Nacional de México,
Instituto Tecnológico de León, León, Guanajuato,
México

² Tecnológico Nacional de México,
Instituto Tecnológico Superior de Purísima del Rincón,
División de Ingeniería Informática, Guanajuato,
México

321ctorres@gmail.com, 14240803@leon.tecnm.mx

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. Este método se aplicó a la base de datos JAFFE, obteniendo un porcentaje de clasificación promedio del 97.78 %.

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. In the area of Web application development there are methodologies as UWE (UML Web Engineering) [1] and W2000 [2] among others, however, it is common practice for developers skip their use and perform in empirical way rather than orderly the development of the applications; when a methodology is used, usually its models are built based on the Unified Modelling Language (UML), which implies higher level of knowledge thereof for proper application of methodologies based on it. Therefore, in this paper are proposed a series of artifacts based on the use of graphs and set theory to design the conceptual modeling of Web applications to achieve the desired results reliably and expeditiously by the developer.

Keywords: Face emotion recognition, image 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.

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

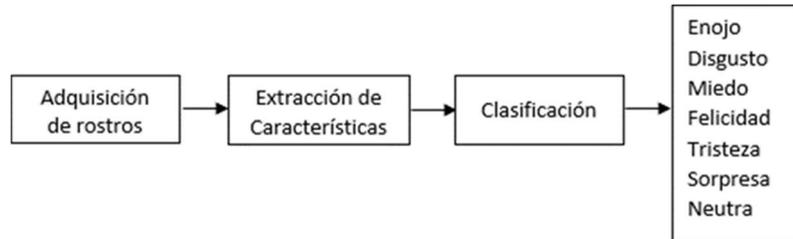


Fig. 1. Sistema de reconocimiento de expresiones faciales [3].

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 (JAFFE). 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 (GE): 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 (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 '<' 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 [11].

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

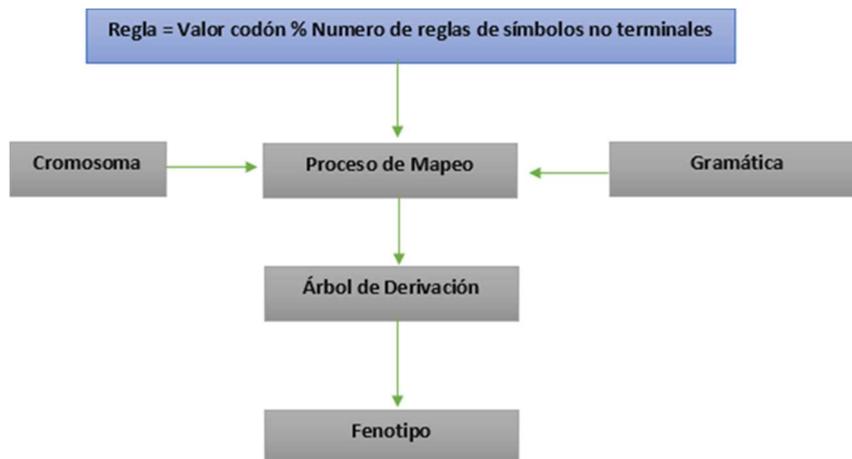


Fig. 2. Diagrama del proceso de mapeo [11].

recibe un codón para aplicar la regla a la gramática e ir formado el árbol de derivación hasta obtener el fenotipo.

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.

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 $\langle signo \rangle\langle OpReal \rangle\langle signo \rangle\langle OpReal \rangle i$.

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.

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). 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 les

```

<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.

Cromosoma	15	3	31	25	61	48	7	5	12	16	3	17	36	22	41	18
-----------	----	---	----	----	----	----	---	---	----	----	---	----	----	----	----	----

(<oper>)<op>(<oper>)	15%1=0
(<signo> <OpReal> <signo> <OpReal> i)<op>(<oper>)	3%1=0
(-<OpReal> <signo> <OpReal> i)<op>(<oper>)	31%2=1
(-<numero> <signo> <OpReal> i)<op>(<oper>)	25%2=1
(-2 <signo> <OpReal> i)<op>(<oper>)	61%10=1
(-2 + <OpReal> i)<op>(<oper>)	48%2=0
(-2 + <numero> i)<op>(<oper>)	7%2=1
(-2 + 6 i)<op>(<oper>)	5%10=5
(-2 + 6 i) + (<oper>)	12%4=0
(-2 + 6 i) + (<signo> <OpReal> <signo> <OpReal> i)	16%1=0
(-2 + 6 i) + (-<OpReal> <signo> <OpReal> i)	3%2=1
(-2 + 6 i) + (-<numero> <signo> <OpReal> i)	17%2=1
(-2 + 6 i) + (-7 <signo> <OpReal> i)	36%10=6
(-2 + 6 i) + (-7 + <OpReal> i)	22%2=0
(-2 + 6 i) + (-7 + <numero> 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.

conoce como individuo y al conjunto de individuos se les denomina población [12]. En la Fig. 5 se muestra el algoritmo de DE.

Histograma de Gradientes Orientados (HOG): es un descriptor que utiliza la magnitud y la orientación del gradiente en porciones localizadas de una imagen denominadas “celdas”; estas cantidades son concatenadas construyendo un histograma de una dimensión tomando en cuenta los niveles de gris de los píxeles y usando las ecuaciones (1) y (2) [5, 13]:

```

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, r2}
    vi ← xr1 + F(xr2 - xr3) vector mutado
    Jr ← entero aleatorio ∈ [1, n]
    Repetir por cada dimensión para j desde 1 hasta n
      rej ← numero aleatorio ∈ [0,1]
      Si (rej < C) o (j == J)
        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].

$$\text{Magnitud: } |\Delta f| = \sqrt{G_x^2 + G_y^2}, \quad (1)$$

$$\text{Orientación: } \theta = \tan^{-1}\left(\frac{G_y}{G_x}\right), \quad (2)$$

donde G_x corresponde al gradiente en el eje horizontal y G_y es el gradiente vertical.

La información es acumulada en un número N de bins (cantidad de elementos a tomar en cuenta por histograma) para posteriormente crear el histograma (ver Fig. 6); el vector de características se forma al concatenar los histogramas creados.

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.

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 creados a partir de una distribución uniforme. 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 selecciones 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 este trabajo se utilizó un determinado número de generaciones).

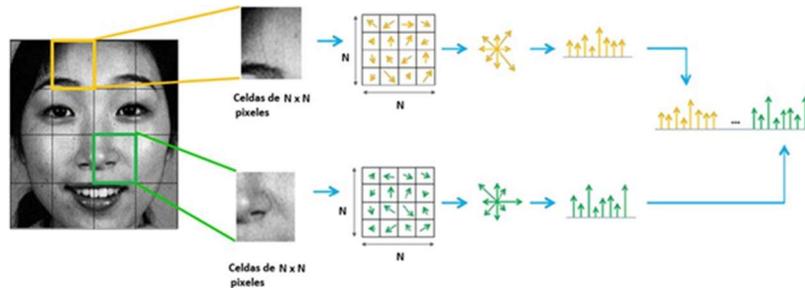


Fig. 6. Caracterización de una imagen con HOG.

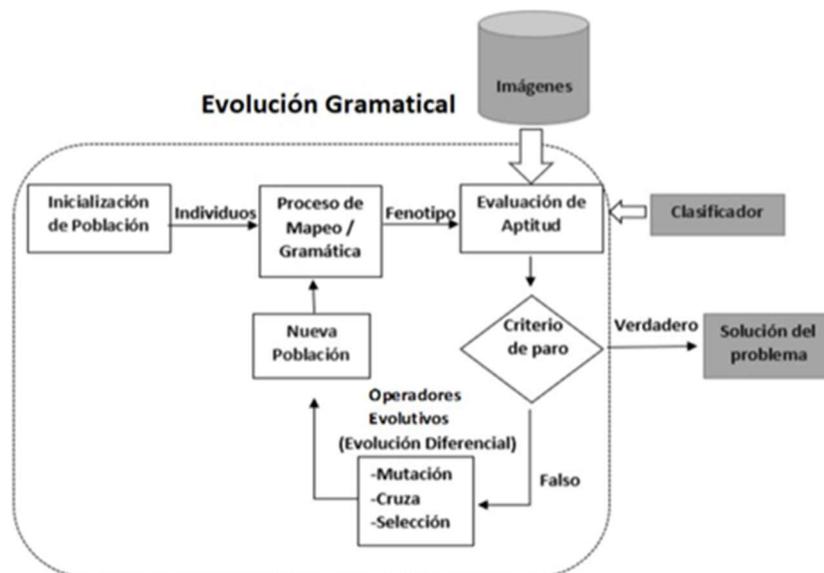


Fig. 7. Diagrama de la metodología propuesta [4].

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.

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 $\langle Start \rangle$.

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

Tabla 1. Operadores básicos utilizados de la Gramática BNF.

Tipo de operador	Nombre	Símbolo 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	Ecuación 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

descriptor a la base de datos JAFFE y se obtiene un nuevo conjunto de imágenes procesadas del mismo tamaño que la base de datos original.

Luego, para cada imagen procesada (del nuevo conjunto de imágenes) 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. El valor de aptitud de los fenotipos más aptos se aproxima al 100 % por lo que a la función se le considera como una función de maximización.

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” que mediante notación postfija se va aplicando paso por paso y de izquierda a derecha a la imagen de prueba, obteniendo una imagen procesada. A

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

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

la imagen resultante se le aplica HOG para obtener un vector de características. Este proceso se repite para cada una de las imágenes de la base de datos JAFFE.

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. 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]. Número de generaciones del proceso evolutivo = 80. Algoritmo HOG: Tamaño de celdas = 32×32 píxeles y tamaño de histogramas = 5 bins. Algoritmo clasificador (SVM): El algoritmo SVM se implementó utilizando la librería Waikato Environment for Knowledge Analysis (WEKA) empleando su configuración original (Kernel Lineal de grado 1, valor $C = 1$, tolerancia de 0.001).

En el proceso de clasificación se utilizó validación cruzada con $K = 10$ pliegues para obtener un porcentaje de clasificación promedio como valor de aptitud de cada fenotipo generado por GE. Para probar la metodología propuesta y tener consistencia en los resultados obtenidos, se realizaron 35 experimentos de manera independiente. En cada

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 Convolutacional	76.56 %
Nuestra propuesta	Evolución Gramatical+ HOG, SVM	97.78 %

experimento se generó inicialmente de manera aleatoria una población de cromosomas, la cual fue modificada a través del proceso evolutivo. En la última generación, se seleccionó al cromosoma más apto (con mayor porcentaje de clasificación), el cual fue el representante de ese experimento. Los parámetros de DE, HOG y SVM se mantuvieron sin cambios a través de todos los experimentos realizados.

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

En esta tabla se puede observar que nuestra metodología usando GE obtuvo resultados semejantes a los de Deep Learning [7]. La realización de un análisis estadístico para comparar el rendimiento de esta propuesta con respecto a las del estado de arte no fue factible debido a que cada metodología propone un diseño experimental diferente, sin embargo, se obtuvo información acerca de los resultados que se han obtenido en relación al reconocimiento de expresiones faciales empleando el dataset JAFFE.

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

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 JAFFE. Evolución Gramatical utilizó una gramática en la forma 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

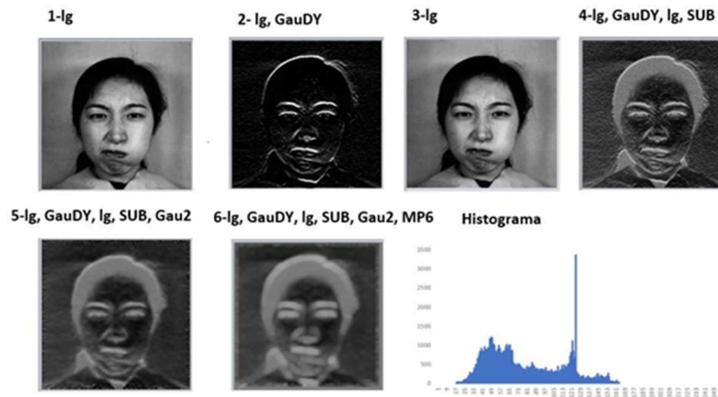


Fig. 8. Ejemplo del proceso de caracterización de una imagen.

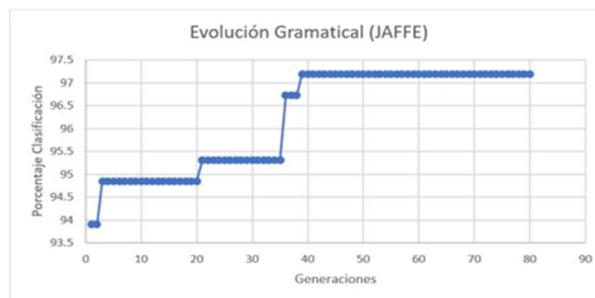


Fig. 9. Gráfica de evolución del mejor valor de aptitud por generación.

mediante una Máquina de Vector Soporte. Los resultados obtenidos mostraron que, para el caso de la base de datos JAFFE, la metodología propuesta presentó un desempeño comparable con metodologías reportadas en el 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 la 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.

Agradecimientos. Los autores agradecen al Tecnológico Nacional de México el apoyo proporcionado. M. A. Torres Fonseca y A. Rojas Domínguez agradecen a CONACYT el apoyo mediante la beca para estudios de posgrado y la beca de investigación CATEDRAS-2598, respectivamente.

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 , IEEE, pp. 1-5 (2018) doi: 10.1109/IWSSIP.2018.8439631
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) doi: 10.1007/s11042-020-09663-2
4. Calzada-Ledesma, V., Puga-Soberanes, H. J., Ornelas-Rodríguez, M., Rojas- Domínguez, A., Carpio-Valadez, J. M., Espinal, A., Sotelo-Figueroa, M. A.: Evolutionary design of problem-adapted image descriptors for texture classification. *IEEE*, vol 6, pp. 40450-40462 (2018) doi: 10.1109/ACCESS.2018.2858660
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 (2019) doi: 10.1088/1757-899X/705/1/012031
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 (2018) doi: 10.1007/978-3-319-97909-0_29
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) doi: 10.1016/j.procs.2017.10.038
8. Melaugh, R., Siddique, N., Coleman, S., Yogarajah, P.: Facial expression recognition on partial facial sections. In: *11th International Symposium on Image and Signal Processing and Analysis*, IEEE, pp. 193-197 (2019)
9. Lyons, M. Kamachi, M. Gyoba J.: Japanese female facial expression database of digital images. (1997) <https://zenodo.org/record/3451524#.YHck3-gzZPY>
10. Moore, J. H., Sipper, M.: Grammatical evolution strategies for bioinformatics and systems genomics. *Handbook of Grammatical Evolution*, Springer, pp. 395-405 (2018) doi:10.1007/978-3-319-78717-6_16
11. Quiroz-Ramírez, O., Espinal, A., Ornelas-Rodríguez, M., Rojas-Domínguez, A., Sánchez, D., Puga-Soberanes, H., Carpio, M., Mancilla Espinoza, L. Ortiz-López, J.: Partially-connected artificial neural networks developed by grammatical evolution for pattern recognition problems. *Fuzzy Logic Augmentation of Neural and Optimization Algorithms: Theoretical Aspects and Real Applications*, *Studies in Computational Intelligence* Springer, vol. 749, pp. 99-112 (2018) doi: 10.1007/978-3-319-71008-2_9
12. Simon, D.: *Evolutionary optimization algorithms*. John Wiley & Sons, pp. 293–305 (2013)
13. Manas, K. B.: *Computer vision and image processing*. Taylor and Francis Group, pp.198–200 (2019)