

# Evolución de descriptores estadísticos de superficie de imágenes por programación genética para el reconocimiento de imágenes por CBIR: una primera aproximación

Héctor Alejandro Tovar Ortiz<sup>1</sup>, César Augusto Puente Montejano<sup>1</sup>,  
Juan Villegas-Cortez<sup>2</sup>, Carlos Avilés Cruz<sup>2</sup>

<sup>1</sup> Universidad Autónoma de San Luis Potosí,  
Facultad de Ingeniería, San Luis Potosí, SLP,  
México

<sup>2</sup> Universidad Autónoma Metropolitana, Azcapotzalco,  
Departamento de Electrónica, Ciudad de México,  
México

tovarha12@gmail.com, cesar.puente@uaslp.mx, {juanvc, caviles}@azc.uam.mx

**Resumen.** El crecimiento sostenido de la Internet en nuestra sociedad se vuelve más evidente al considerar que más de la mitad de la información en ella son imágenes, y el gran problema por atacar sigue siendo el reconocimiento de imágenes de forma automática y no supervisada. La metodología de reconocimiento de imágenes a partir de su contenido propio (CBIR) ha mostrado tener mucha efectividad, tratando de simular el comportamiento de clasificación natural de las imágenes como los seres humanos lo realizamos: verla, analizarla y determinar su clasificación. En este trabajo presentamos la solución de crear nuevos descriptores estadísticos que apoyen la metodología CBIR, aplicados sobre imágenes de escenarios naturales, por medio de la Programación Genética, a partir de un conjunto de funciones intuitivo, y con el alto costo computacional de considerar a la función de aptitud como un clasificador de distancia mínima y auto organización de racimos de atributos, logrando hasta este momento, una clasificación y reconocimiento de casi el 100 %.

**Palabras clave:** Algoritmos evolutivos, programación genética, reconocimiento de imágenes, CBIR, reconocimiento de patrones.

## CBIR Pattern Recognition Using Genetic Programming: First Approximation Using New Descriptors for Images Textures

**Abstract.** The steady increase of Internet in our society becomes more evident against the crude fact that more than a half of the internet infor-

mation consists of images. This situation poses an open problem, how to recognize images automatically and unsupervised. The methodology for recognizing images by means of their own content (CBIR) has shown a lot of effectivity. CBIR resembles the orderly way in which humans manage to recognize images: seeing, analyzing and then classify them. In this work we present a solution to this problem by creating new statistical descriptors supporting the CBIR methodology. These are applied over images of natural scenes by means of Genetic Programming using a set of intuitive functions. This process involves the use of an amplitude function as a classifier of the self organization and the minimum distance of the attribute clusters. The incorporation of this last feature comes with a high computational cost. However, up to date, our results get a classification and recognition of almost 100 %.

**Keywords:** Evolutionary algorithms, genetic programming, image recognition, CBIR, pattern recognition.

## 1. Introducción

En los últimos años el crecimiento sostenido de la Internet ha impactado diferentes áreas del crecimiento humano, entre ellos esta el conocimiento compartido por medio de la información que en ella se dispone, y acorde a [6] más de la mitad de ésta es en imágenes. El problema de manejar imágenes en grandes cantidades, como a las que hemos llegado con el Big Data, es que éstas no están nombradas acorde a su contenido, y por un lado se requiere poder apoyarse en las imágenes para diferentes propósitos de discurso, y por el otro las mismas imágenes en cantidad van creciendo; por ello se sigue trabajando en la comunidad mundial del área científica en poder desarrollar nuevas formas de su almacenamiento, distribución, alta disponibilidad y su clasificación acorde a diferentes propósitos. Los seres humanos podemos de forma simple clasificar un conjunto de imágenes, es algo que hacemos cotidianamente, pero lograr que una computadora lo haga no es tarea sencilla, y en esta tarea se centra el trabajo aquí presentado, en cómo mejorar una de las técnicas de clasificación automática de imágenes, que parte del contenido propio de cada imagen obtenido tras su análisis de superficie, llamada CBIR (Content Based Image Retrieval) [6].

El Cómputo Evolutivo ha mostrado brindar nuevas soluciones para problemas que han sido resueltos de forma determinística, e.g., los problemas de programación no lineal, o de optimización; específicamente en este trabajo hemos aplicado la metodología evolutiva de la Programación Genética, como herramienta fundamental que nos permite la exploración de nuevos programas-individuos de una población, y su desempeño como posible solución al problema formulado por cada uno de los programas evolucionados. Es así, que con la metodología planteada se ha logrado, hasta este momento, hallar nuevas soluciones que resuelven un problema muy complejo, haciendo variaciones sobre las imágenes usadas para el entrenamiento de un clasificador, brindando soluciones factibles.

En la Sección 2 se muestra el estado del arte de nuestro alcance, en la Sección 3 compartimos la metodología planteada de CBIR+PG, en la Sección 4 presen-

tamos la experimentación y los resultados obtenidos, y finalmente compartimos las conclusiones alcanzadas en la Sección 5.

## 2. Estado del arte

La disciplina de la recuperación de imágenes a partir del contenido de las mismas, o CBIR, es una de las más activas, considerando que ha ido madurando desde la fecha clave de febrero de 1992, donde la US National Science Foundation (USNSF) organizara una primera reunión de trabajo en Redwood, California, para identificar las áreas de investigación principales que deberían considerarse en cuenta por los investigadores para la administración de los sistemas de información visual, mismo que pudieran ser de utilidad para la ciencia, industria, medicina, medio ambiente, educación, entretenimiento y otras aplicaciones [9]; posteriormente a ésta reunión el navegador de Internet Mosaic fue liberado, iniciando una revolución en la web que iría cambiando rápidamente, y en ese tiempo también se fueron desarrollando nuevos foto sensores y haciéndose disponibles; con todo esto, el número de imágenes almacenadas comenzó a incrementarse y la necesidad de crear herramientas que ayudaran a su ordenamiento, clasificación e indexación fue algo prioritario a desarrollar [8].

Usando CBIR se ha tratado de poner un orden las imágenes, a fin de poder lograr su clasificación para una rápida identificación. Actualmente se han desarrollado aplicaciones con un alto nivel de reconocimiento tanto para imágenes en escenarios naturales [6], así como para identificar rostros humanos [1], con la particularidad en ésta última aplicación a rostros de que se considera la información de la “textura” del rostro de la persona, y a partir de éstas características se realiza la clasificación. Una metodología general de CBIR se muestra a la derecha de la figura 1, en ese se aprecia cómo a partir de un conjunto de imágenes de entrenamiento se lee cada imagen, se cambia al espacio de color HSI, donde se tiene más información no perceptible a nuestros ojos (que sólo vemos en el espacio de color RGB), luego sobre la nueva imagen se realiza un sembrado de puntos aleatorios, y en cada punto se abre una ventana de tamaño  $10 \times 10$  píxeles, y sobre ésta ventana se extraen una tripleta de características estadísticas. En [6] y [1] se muestra el uso de tres descriptores de superficie: la media, la desviación estándar y la homogeneidad (obtenida a partir de los valores de la matriz de co-ocurrencia) [3]; y es con esta tripleta de valores que se logra hacer la caracterización de las imágenes a partir de un análisis local, extrapolándose hacia toda la imagen.

Los Algoritmo Evolutivos (AE) han mostrado ser de utilidad en brindar soluciones factibles, para problemas debidamente formulados con restricciones [2]; y hablando sólo de la posibilidad de acoplar un algoritmo genético (AG) para una mejor disposición del número de racimos dispuestos para aglomerar los patrones obtenidos se tiene en [5]. Como parte de los AE tenemos a la Programación Genética (PG), que a diferencia de loa AG, no se limitan en su búsqueda evolutiva a una cadena genómica de longitud finita, como se formulan inicialmente los AG; la PG se basa en la evolución de programas completos, don-

de cada uno es una solución candidata al problema, siendo ésta su formulación fundamental dada por J. Koza [4].

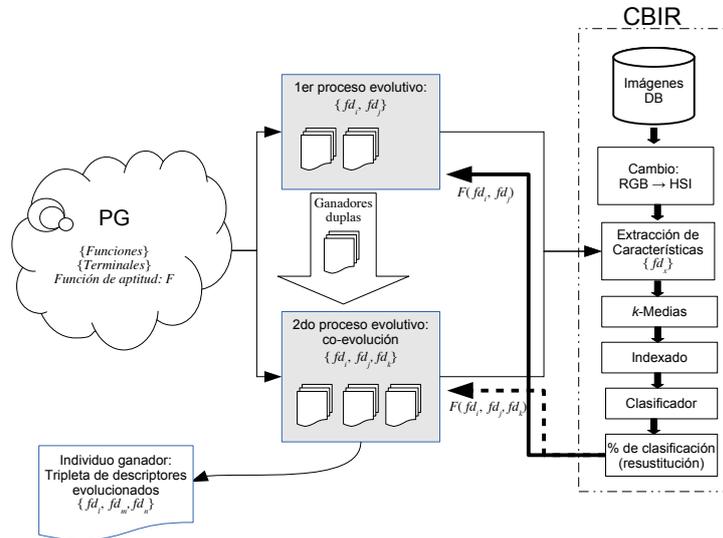
En este trabajo proponemos la aplicación de la coevolución, desde el principio de “divide y vencerás” de la formulación del modelo del problema a resolver, donde cada población depende de la otra, o se dice que tiene una relación estrecha. Los modelos de coevolución básicos son: cooperativa y presa-cazador, el mejor ejemplo de la primera son las colonias de hormigas, donde entre todos los individuos de la población buscan cooperar para hallar una solución al problema, mientras que en el segundo modelo, una parte es la que busca sobrevivir, la presa o el cazador, pero si una de las dos especies muere, la otra también lo hará, de ahí que se tiene que llegar a un equilibrio entre las especies. Dado lo anterior, en este trabajo formulamos una coevolución cooperativa, inspirada en el modelo de Memorias Asociativas Evolutivas (MAE) [10], con la finalidad de proponer nuevos descriptores locales de superficies de imágenes para ser usados desde la metodología CBIR.

### 3. Metodología

Nuestra solución propuesta es la combinación de CBIR con PG para identificar imágenes de escenarios naturales, y para esta primera aproximación usamos la base de imágenes de J. Voguel [11], con la metodología CBIR reportada en [6] y mejorada en [1], misma que se detalla en general sobre el lado derecho de la figura 1. La aplicación de PG se realiza en evolucionar el conjunto de descriptores como una tripleta, cada individuo entonces se coevoluciona en dos fases, que son detalladas más abajo; siendo en la primera la evolución de una pareja de descriptores, y los ganadores de al menos  $N$  procesos se aportarán en el segundo proceso o fase, para lograr el tercer descriptor, repitiéndose  $N$  veces, para al final de todo el proceso tener una tripleta de descriptores de la ventana de superficie de la imagen a estudiar.

El algoritmo se divide en dos fases (procesos evolutivos 1 y 2), que se explican a continuación. Durante el primer proceso evolutivo se buscan los primeros dos elementos de la tripleta final, mediante un algoritmo evolutivo de PG. En el segundo proceso evolutivo, se toman los pares generados en la fase anterior y se utilizan para sintetizar tripletas mediante otro algoritmo evolutivo de PG. Los parámetros de la PG considerados como fundamentales para los dos procesos evolutivos considerados son: el conjunto de funciones  $F_a$ , el conjunto de terminales  $T_a$  y la función de aptitud  $F$ , tal como se describen a continuación:

- $F_a := \{+, -, *, mydivide, sen, cos\}$ ,
- $T_a : \{\text{El conjunto de descriptores estadísticos: } fd01, fd02, \dots, fd09\}$ ,
- $F$  : la función de aptitud : el porcentaje de recuperación sobre toda la base de datos (DB) considerada, entrenando el clasificador sobre CBIR con los descriptores evolucionados, éste valor es usando la técnica de resustitución, i.e., se entrena y prueba el clasificador usando toda la DB. Aquí consideramos el óptimo alcanzar el 100 %, acorde con la ecuación 1, tal que  $f \in [0, 1]$ , siendo el mejor valor 1.



**Fig. 1.** Metodología general propuesta de evolución de descriptores de textura de imágenes, en coevolución cooperativa, usando CBIR

$$F = \frac{\text{Num de imágenes recuperadas}}{\text{Num de imágenes totales}}. \quad (1)$$

Los descriptores considerados se calculan sobre cada ventana por capa, para esta primera aproximación evolutiva que estamos reportando, y son los siguientes:

- $fd01 : fd1_H = \text{media}$ ,
- $fd02 : fd2_H = \text{desv. std}$ ,
- $fd03 : fd3_H = \text{homogeneidad-coocurrencia}$ ,
- $fd04 : fd4_S = \text{media}$ ,
- $fd05 : fd5_S = \text{desv. std}$ ,
- $fd06 : fd6_S = \text{homogeneidad-coocurrencia}$ ,
- $fd07 : fd7_I = \text{media}$ ,
- $fd08 : fd8_I = \text{desv. std}$ ,
- $fd09 : fd9_I = \text{homogeneidad-coocurrencia}$ .

Es dentro de la evaluación de la función de aptitud,  $F$ , donde con base a los descriptores evolucionados en cada proceso evolutivo (1 y 2), que se toma el análisis CBIR sobre toda la base de imágenes que en resumen por cada punto aleatorio sobre la imagen se abre una ventana de  $10 \times 10$  píxeles, quedando 3 ventanas por punto (una ventana por capa en espacio de color HSI), y sobre cada ventana considerada por separado se calcula el descriptor elegido, formando un

patrón con 3 entradas por descriptor por ventana, tal que si son 3 capas y 3 descriptores, el patrón queda de dimensión  $1 \times 9$ , se pasan todos los patrones así formados a organizarse por medio del algoritmo  $K$ -medias, y finalmente aplicando un clasificador de distancia  $k$ -NN, se toman a los  $k$  vecinos más próximos. Más detalle se muestra en [6]. Un aspecto importante de la consideración del cambio de espacio de color a HSI, es que se tiene información de textura de la imagen y no del color. A continuación se describe detalladamente cada una de estas fases.

**Fase 1.** Primero, se configuran los parámetros del algoritmo de programación genética. Después, se genera la población de soluciones inicial para el proceso evolutivo. Cada solución se forma por un par de elementos. El primer elemento representa la combinación matemática de uno o más de los descriptores mencionados antes. Este elemento es sintetizado por el algoritmo, utilizando el conjunto de funciones y terminales que se ilustran más arriba; mientras que el segundo elemento representa un solo descriptor seleccionado al azar de entre los 9 propuestos. Posteriormente se evalúa el desempeño de cada una de estas soluciones, mediante el proceso de reconocimiento de imágenes a partir de su propio contenido que se propuso en el trabajo publicado en [6]. Las soluciones mejor evaluadas son seleccionadas para llevar a cabo la generación de la nueva población mediante los métodos de cruzamiento y mutación. Es necesario mencionar que la mejor solución se pasa directamente a la nueva población. Este proceso se lleva a cabo por un determinado número de generaciones, hasta que la mejor solución de la población no cambie. Una vez que el proceso evolutivo se detiene, se guarda la mejor solución para un uso posterior en la fase 2. La fase 1 se lleva a cabo 3 veces, obteniendo así 3 soluciones ganadoras de cada ejecución de esta fase.

**Fase 2.** El proceso evolutivo es similar al de la fase 1. En esta caso la población de soluciones se conforma de un solo elemento. Este elemento es sintetizado por el algoritmo evolutivo de esta fase, utilizando el conjunto de funciones y terminales que se ilustran en la Tabla 1. El proceso de evaluación en esta fase consiste nuevamente en aplicar el proceso de reconocimiento de imágenes a partir de su propio contenido que se propuso en el trabajo de [6]. La diferencia es, que se fabrican 3 tripletas cada vez. Una por cada una de las soluciones ganadoras de la fase previa; añadiendo a cada una el elemento sintetizado en esta fase. Por lo tanto, se obtienen tres evaluaciones por cada individuo; así que se conserva la mejor evaluación y se registra la tripleta que obtuvo dicha evaluación para que, al término del proceso evolutivo se conozca la tripleta fabricada con el individuo ganador. Esta tripleta representa la solución al problema.

## 4. Experimentos y resultados

Los experimentos consistieron en probar el algoritmo que se detalló en la sección anterior sobre la imágenes de la base de datos de paisajes naturales desarrollada en [11], con resolución de  $720 \times 480$  píxeles. De esta base de datos



**Fig. 2.** Ejemplos de las imágenes de los 5 escenarios naturales considerados, las pruebas se hicieron sobre dos tipos o clases

se tomaron 2 clases de imágenes, con 20 imágenes por clase como se muestra en la Figura 2. La metodología descrita en el capítulo anterior se implementó utilizando MatLab©, con el módulo de PG conocido como GPLab, el cual fue desarrollado en [7]. La Tabla 1 muestra el valor de los parámetros utilizados para el algoritmo de GP durante las pruebas experimentales. Los valores adecuados para cada parámetros se determinaron mediante prueba y error.

**Tabla 1.** Parámetros utilizados para el algoritmo basado en GP. Dichos parámetros están ordenados de acuerdo al módulo de GPLab al que pertenecen

<b>Módulo Población inicial</b>	
Población inicial	Mitad árboles completos y Mitad árboles de distribución aleatoria
<b>Módulo Generación</b>	
Generaciones	300
Tamaño de la población	100 individuos
Probabilidad de cruzamiento	0.7
Probabilidad de mutación	0.3
Tipo de Selección	Torneo de presión lexicográfica
Tipo de elitismo	Conservación del mejor por generación
<b>Módulo Manejo de los individuos</b>	
Profundidad de los árboles	Dinámica
Profundidad máxima dinámica	3 niveles
Profundidad máxima real	4 niveles

#### 4.1. Análisis de resultados y comparación con otros resultados

Se realizaron 3 corridas. Los resultados que se hallaron de clasificación para cada una de estas corridas se muestran en la Tabla 2. Se puede apreciar que con los ajustes evolutivos para la PG se ha logrado un nivel de clasificación cercano al 100% sobre 2 clases de la DB, 97.13% y 96.63%, con una complejidad de operadores bastante sencilla como los mostrados en las dos tripletas  $\{descriptor_1, descriptor_2, descriptor_3\}$ , con apenas un espacio de exploración de funciones de 6 tipos, con una profundidad y amplitud de árboles controladas.

**Tabla 2.** Soluciones encontradas en cada una de las tres corridas que se llevaron a cabo. Se presentan los árboles sintácticos de cada uno de los descriptores que conforman de la tripleta, así como el porcentaje de clasificación que alcanzan

Corrida	Descriptor 1	Descriptor 2	Descriptor 3	Desempeño obtenido
1				<b>97.13 %</b>
2				<b>96.63 %</b>

Comparando con los resultados obtenidos en [5,6], estamos muy próximos a igualarlos por menos del 3%, y por encima de lo reportado por [11], pero considerando la mejora del clasificador de distancia por análisis local de textura. En cada uno de los 3 trabajos mencionados anteriormente, los descriptores fueron construidos manualmente acorde a la experiencia del procesamiento digital de imágenes, pero en nuestra metodología evolutiva propuesta estamos hallando nuevas soluciones en descriptores que no se habían considerado anteriormente en la literatura, con la limitante de la programación estructurada.

## 5. Conclusiones y trabajo futuro

En este artículo hemos presentado una primera aproximación, hasta ahora no documentada, de aplicar la evolución de descriptores estadísticos por medio

de la PG hacia la metodología CBIR, aplicada al problema de clasificación de escenarios naturales. Hasta este momento la exploración ha dado un resultado que podemos considerarlo muy bueno tras resolver el reto de la programación del paradigma de la coevolución desde una plataforma que per sé no la permite sobre un algoritmo evolutivo, dado el fenómeno de *bloat* documentado en la literatura, alcanzado unos descriptores robustos para dos escenarios naturales de 5 clases, que si vemos en la literatura, el problema no se resuelve satisfactoriamente por segmentación de las imágenes, pero sí por el análisis del contenido de las imágenes por medio de CBIR.

En un trabajo futuro se considera la coevolución con descriptores estadísticos que consideren la transformación hacia otros espacios de color, o bien de ser posible tomar otras bandas, e.g., el infrarrojo, para un problema en específico, con la idea de poder explorar evolutivamente soluciones más robustas para usar una técnica probada de clasificación de imágenes, pero sin importar el tipo de imágenes. Adicionalmente, será necesario trasladar la programación hacia un lenguaje de programación de bajo nivel, como ANSI C/C++, a fin de lograr mayor rapidez y precisión; y todo esto para lograr probar sobre más clases de imágenes.

## Referencias

1. Benavides-Alvarez, C., Villegas-Cortez, J., Roman-Alonso, G., Aviles-Cruz, C.: Reconocimiento de rostros a partir de la propia imagen usando tecnica cbir. In: X Congreso Espanol sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados MAEB 2015. pp. 733–740. Universidad de Extremadura, Merida, Extremadura. Spain (Jan 2015)
2. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley Publishing (2001)
3. Fukunaga, K.: Introduction to statistical pattern recognition (2nd ed.). Academic Press Professional, Inc., San Diego, CA, USA (1990)
4. Koza, J.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press (1992)
5. Perez-Pimentel, Y., Osuna-Galan, I., Villegas-Cortez, J., Aviles-Cruz, C.: A genetic algorithm applied to content-based image retrieval for natural scenes classification. In: 13th Mexican International Conference on Artificial Intelligence (MICAI). pp. 155–161 (Nov 2014)
6. Serrano-Talamantes, J.F., Aviles-Cruz, C., Villegas-Cortez, J., Sossa-Azuela, J.H.: Self organizing natural scene image retrieval. Expert Systems with Applications 40(7), 2398–2409 (2013), <http://www.sciencedirect.com/science/article/pii/S0957417412011888?v=s5>
7. Silva, S., Almeida, J.: Gplab-a genetic programming toolbox for matlab. In: In Proc. of the Nordic MATLAB Conference (NMC-2003). pp. 273–278 (2005)
8. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(12), 1349–1380 (Dec 2000)
9. USNSF: US NSF workshop visual information management systems workshop report. In: Conference on Storage and Retrieval for image and Video Databases. Springer Secaucus, Denver, Colorado, USA (1992)

10. Villegas-Cortez, J., Olague, G., Aviles, C., Sossa, H., Ferreyra, A.: Automatic synthesis of associative memories through genetic programming: A first co-evolutionary approach. In: Applications of Evolutionary Computation, Lecture Notes in Computer Science, vol. 6024, pp. 344–351. Springer Berlin / Heidelberg (2010)
11. Vogel, J., Schiele, B.: Semantic modeling of natural scenes for content-based image retrieval. *International Journal of Computer Vision* 72(2), 133–157 (2006), <http://dx.doi.org/10.1007/s11263-006-8614-1>