

**Centro de Investigación Científica y de Educación  
Superior de Ensenada, Baja California**



**Programa de Posgrado en Ciencias  
en Ciencias de la Computación**

---

**Corteza visual evolutiva para la clasificación de objetos en  
imágenes naturales usando un enfoque multi-objetivo**

Tesis

para cubrir parcialmente los requisitos necesarios para obtener el grado de  
Doctor en Ciencias

Presenta:

**Daniel Eduardo Hernández Morales**

Ensenada, Baja California, México

2016

Tesis defendida por

**Daniel Eduardo Hernández Morales**

y aprobada por el siguiente Comité

---

Dr. Gustavo Olague Caballero  
Director del Comité

Dr. Jesús Favela Vara

Dr. Vitaly Kober

Dr. Benjamín Hernández Valencia



---

Dr. Jesús Favela Vara  
Coordinador del Programa de Posgrado en Ciencias de la Computación

---

Dra. Rufina Hernández Martínez  
Director de Estudios de Posgrado

Resumen de la tesis que presenta Daniel Eduardo Hernández Morales como requisito parcial para la obtención del grado de Doctor en Ciencias en Ciencias de la Computación.

## **Corteza visual evolutiva para la clasificación de objetos en imágenes naturales usando un enfoque multi-objetivo**

Resumen aprobado por:

---

Dr. Gustavo Olague Caballero

Director de tesis

En la actualidad, los sistemas de visión por computadora se han inspirado del funcionamiento del sistema visual humana con el objetivo de resolver diferentes problemas, tales como la detección y clasificación de objetos. Modelos computacionales como el modelo de la corteza visual artificial han mostrado resultados prometedores en resolver ese tipo de problemas. Por lo tanto, este trabajo de tesis propone una nueva metodología para la creación de un vector descriptor de imágenes con el objetivo de clasificarlas; y al mismo tiempo, encontrar la posición en la imagen del objeto de interés. Además, este trabajo implementa el paradigma de optimización conocido como programación de cerebros con un enfoque multi-objetivo; en busca de integrar las tareas visuales de localización y descripción de objetos, con el objetivo de mejorar el rendimiento del sistema en el problema de clasificación de objetos. Dicha metodología se implementa para generar soluciones, en base al modelo propuesto, para clasificar las imágenes de dos bases de datos (GRAZ-01 y GRAZ-2) utilizadas en el estado-del-arte como punto de referencia para comparar el rendimiento de diferentes sistemas de clasificación. Los modelos generados por medio de la programación cerebral presentes en este trabajo de investigación logran resultados similares, y en algunos casos superiores, a otras técnicas del estado-del-arte en la tarea de clasificación de las imágenes previamente mencionadas.

Palabras Clave: **Corteza visual artificial, clasificación de objetos, detección de objetos, evolución de cerebros multi-objetivo**

Abstract of the thesis presented by Daniel Eduardo Hernández Morales as a partial requirement to obtain the Doctor of Science degree in Computer Science.

## **Evolutionary multi-objective visual cortex for object classification in natural images**

Abstract approved by:

---

Dr. Gustavo Olague Caballero

Thesis director

In recent years, computer vision systems have used the human visual system as inspiration for solving different tasks such as object detection and classification. Along these lines, computational models such as the Artificial Visual Cortex (AVC) have shown promising results while performing difficult visual tasks. In this way, this thesis proposes a new methodology for creating an image descriptor vector in order to classify it, and simultaneously finding the objects' location within the image. This work implements the brain programming paradigm from a multi-objective perspective; seeking to integrate two visual tasks, object localization and object description, in the interest of improving the system's performance in the object classification task. This methodology is implemented for training the proposed model for classify the images from the GRAZ-01 and GRAZ-02 databases, which serve as a comparison standard for classification systems. The solutions found in this research match, and in some cases outperform, other techniques of the state-of-the-art for classifying the aforementioned databases.

Keywords: **Artificial Visual Cortex, Object Classification, Object Detection, Multi-objective Brain Programming**



## Dedicatoria

***“La vida es la cosa mejor que se ha inventado”.***

***A mis padres.***

***A Adriana.***

## Agradecimientos

“Cuando la gratitud es tan absoluta las palabras sobran”.

Quiero agradecer a mi esposa Adriana, por decidir acompañarme paso a paso en esta vida, por tu apoyo en cada una de mis aventuras. Pero sobre todo, gracias por tu paciencia, comprensión y amor. Sin ti esto no tendría sentido.

A mis padres, tengo tanto que agradecerles que la vida no me alcanzará para terminar de hacerlo. Gracias por apoyarme a lograr mis sueños, sin su ayuda esto no habría sido posible. Ustedes me han dado las lecciones más importantes, ustedes me enseñaron a vivir. A mis hermanos y el resto de mi familia, gracias por siempre estar ahí y creer en mi.

A mis amigos, “les adeudo la paciencia de tolerarme; los arrebatos de humor, la negligencia, las vanidades, los temores y las dudas”.

A mi asesor, por sus consejos y su guía, que permitieron mi desarrollo científico y humano.

A mis sinodales por sus observaciones y consejos que ayudaron a enriquecer este trabajo.

A mis compañeros de EvoVisión, en especial a Eddie Clemente por su apoyo y amistad.

Al Centro de Investigación Científica y de Educación Superior de Ensenada.

Al pueblo mexicano; quienes, a través del Consejo Nacional de Ciencia y Tecnología (CONACyT), me brindaron el apoyo económico para realizar mis estudios doctorales.

# Tabla de contenido

	Página
<b>Resumen en español</b>	<b>ii</b>
<b>Resumen en inglés</b>	<b>iii</b>
<b>Dedicatoria</b>	<b>iv</b>
<b>Agradecimientos</b>	<b>v</b>
<b>Lista de figuras</b>	<b>viii</b>
<b>Lista de tablas</b>	<b>xii</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Atención visual . . . . .	3
1.2. Planteamiento del problema . . . . .	4
1.3. Objetivo general . . . . .	7
1.4. Objetivos específicos . . . . .	7
1.5. Preguntas de investigación . . . . .	8
1.6. Alcance de la tesis . . . . .	8
1.7. Organización de la tesis . . . . .	9
<b>2. Marco teórico</b>	<b>11</b>
2.1. La evolución como paradigma computacional . . . . .	11
2.1.1. Programación genética . . . . .	14
2.1.2. Evolución multi-objetivo . . . . .	16
2.1.2.1. Optimización multi-objetivo . . . . .	17
2.1.2.2. Algoritmos evolutivos multi-objetivo . . . . .	18
2.2. Operaciones sobre imágenes . . . . .	18
2.2.1. Convolución . . . . .	19
2.2.2. Derivada de una imagen . . . . .	21
2.2.3. Suavizado . . . . .	22
2.2.4. Operación de umbral . . . . .	24
2.2.5. Operaciones punto por punto . . . . .	25
2.2.6. Morfología . . . . .	26
<b>3. Corteza visual artificial multi-objetivo</b>	<b>33</b>
3.1. Antecedentes . . . . .	33
3.1.1. La corteza visual . . . . .	34
3.1.2. Modelo de la ruta dorsal artificial . . . . .	41
3.1.3. Integración de las dos rutas . . . . .	46
3.2. Trabajo relacionado . . . . .	49
3.3. Metodología . . . . .	51
3.3.1. El modelo de la corteza visual artificial multi-objetivo . . . . .	52
3.3.1.1. Detección y transformación de características visuales . . . . .	53
3.3.1.2. Detección y descripción del objeto . . . . .	55
<b>4. Programación de cerebros artificiales con enfoque multi-objetivo</b>	<b>58</b>
4.1. Programación de cerebros multi-objetivo . . . . .	58

4.1.1.	Trabajos previos: Enfoques mono-objetivo . . . . .	58
4.2.	Evaluación multi-objetivo . . . . .	60
4.3.	Genotipo . . . . .	62
4.4.	Operadores genéticos . . . . .	63
4.4.1.	Cruce . . . . .	63
4.4.2.	Opeardores de mutación . . . . .	64
4.4.3.	Funciones y terminales . . . . .	65
<b>5.</b>	<b>Experimentos y resultados</b>	<b>68</b>
5.1.	Base de datos de imágenes . . . . .	69
5.2.	Validación cruzada y comparación con otros métodos . . . . .	70
<b>6.</b>	<b>Conclusiones y trabajo futuro</b>	<b>96</b>
6.1.	Aportaciones de la tesis . . . . .	97
6.2.	Trabajo futuro . . . . .	98
	<b>Lista de referencias bibliográficas</b>	<b>99</b>

## Lista de figuras

Figura		Página
1.	La atención visual se enfoca en los objetos de interés para la tarea que se está realizando. . . . .	4
2.	La construcción de la representación $R_j$ , también conocida como vector descriptor, debe relacionar la clase del objeto, sin importar las transformaciones visuales que pueda sufrir el objeto en la imagen $I$ ; tales como traslación, rotación, escala o cambio de iluminación. . . . .	5
3.	Descripción general del funcionamiento de dos enfoques para resolver problemas. . . . .	12
4.	Estructura básica de un algoritmo de GP. Los individuos representados como árboles, construidos con las primitivas; además, se muestran los mecanismos de variación para la creación de nuevos individuos. Diagrama propuesto originalmente por Trujillo y Olague (2008). . . . .	15
5.	Espacio de decisión que se relaciona al espacio de objetivos por medio del mapeo $f$ . . . . .	16
6.	Vecindario de tamaño $3 \times 3$ alrededor del pixel $(u, v)$ . . . . .	23
7.	Ejemplo de suavizado Gaussiano con $\sigma = 2$ y un núcleo de $5 \times 5$ . . . . .	24
8.	Ejemplo del uso de la operación de umbral de Otsu para transformar una imagen en escala de grises a una imagen binaria. . . . .	25
9.	Ejemplo del uso de la operación morfológica de dilatación usando un elemento estructural en forma de diamante. . . . .	27
10.	Ejemplo del uso de la operación morfológica de erosión usando un elemento estructural en forma de disco. . . . .	28
11.	Ejemplo del uso de la operación morfológica de apertura usando un elemento estructural en forma de cuadro. . . . .	28
12.	Ejemplo del uso de la operación morfológica de cerradura usando un elemento estructural en forma de cuadro. . . . .	29
13.	Ejemplo del uso de la operación morfológica de añadir o eliminar usando un elemento estructural en forma de diamante. . . . .	30
14.	Ejemplo del uso de la operación morfológica de sombrero superior e inferior. . . . .	30
15.	Ejemplo del uso de la operación morfológica de esqueleto. . . . .	31
16.	Ejemplo del uso de la operación morfológica de perímetro. . . . .	31
17.	Esquema del modelo de HMAX. Las capas S realizan operaciones lineales ( <i>template matching</i> ) y las capas C realizan operaciones no lineales (operación <i>max</i> ) (Riesenhuber y Poggio, 1999). . . . .	36

Figura	Página
18. Diagrama a bloques del sistema propuesto por Serre <i>et al.</i> (2007). La imagen de entrada se descompone en 4 direcciones y 16 escalas, se asocian los resultados usando la operación de máximo y se realiza una comparación con cada una de las características previamente aprendidas; finalmente la clasificación se hace a través de una SVM. . . . .	37
19. Diagrama del flujo de información en el modelo propuesto por Mutch y Lowe (2008). . . . .	39
20. Comparación entre el diccionario de parches creado por el modelo HMAX y el modelo expuesto por Ghodrati <i>et al.</i> (2012). . . . .	40
21. Modelo de atención visual propuesto por Koch y Ullman (1985). Este diagrama muestra las etapas de la atención, los mapas de características como representación temprana, seguidos del mapa de sobresalencia, que después del proceso selectivo lleva a una representación central. . . . .	42
22. Modelo para el control de la atención abajo-arriba propuesto por Itti y Koch (2001). . . . .	43
23. Ilustración de los pasos del proceso para obtener las regiones atendidas por el modelo propuesto por Walther y Koch (2006). . . . .	44
24. Analogía entre sistema natural y la ADS. . . . .	45
25. Modelo del AVC propuesto por Clemente (2015). . . . .	48
26. Diagrama de flujo de la información visual en el modelo propuesto AVCMO. . . . .	54
27. Representación gráfica del proceso iterativo para agregar el valor máximo en el vecindario de los elementos de $\vec{p}$ . La región azul son los puntos que forman $\vec{p}$ , mientras que la región verde es el vecindario donde se busca el siguiente elemento. . . . .	56
28. Comparación entre la región de la imagen atendida por el modelo, y la región ocupada por el objeto de interés. Utilizadas para evaluar la precisión $\rho$ y la sensibilidad $\vartheta$ . . . . .	62
29. Los operadores genéticos se implementan en dos niveles; la Figura (a) muestra la operación de cruce a nivel cromosoma, y la (b) a nivel gen; mientras que la (c) y (d) representan la operación de mutación a nivel cromosoma y nivel gen respectivamente. . . . .	63
30. Comparación usando las curvas ROC para la clase Bicicleta de GRAZ-01. . . . .	76
31. Comparación usando las curvas ROC para la clase Persona de GRAZ-01. . . . .	76
32. Comparación usando las curvas ROC para la clase Bicicleta de GRAZ-02. . . . .	77
33. Comparación usando las curvas ROC para la clase Persona de GRAZ-02. . . . .	77
34. Comparación usando las curvas ROC para la clase Carros de GRAZ-02. . . . .	78

Figura	Página
35. Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Bicicleta de GRAZ-01. El individuo señalado corresponde a la solución presente en la Tabla 10. . . . .	79
36. Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Persona de GRAZ-01. El individuo señalado corresponde a la solución presente en la Tabla 10. . . . .	79
37. Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Bicicleta de GRAZ-02. El individuo señalado corresponde a la solución presente en la Tabla 10. . . . .	80
38. Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Persona de GRAZ-02. El individuo señalado corresponde a la solución presente en la Tabla 10. . . . .	80
39. Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Carros de GRAZ-02. El individuo señalado corresponde a la solución presente en la Tabla 10. . . . .	81
40. Gráficas de convergencia del algoritmo de BP para la clase Bicicleta de GRAZ-01. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobles. . . . .	82
41. Gráficas de convergencia del algoritmo de BP para la clase Persona de GRAZ-01. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobles. . . . .	83
42. Gráficas de convergencia del algoritmo de BP para la clase Bicicleta de GRAZ-02. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobles. . . . .	84
43. Gráficas de convergencia del algoritmo de BP para la clase Persona de GRAZ-02. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobles. . . . .	85
44. Gráficas de convergencia del algoritmo de BP para la clase Carros de GRAZ-02. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobles. . . . .	86
45. Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de color, utilizadas para crear las soluciones para las clases de GRAZ-01. . . . .	88

Figura	Página
46. Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de orientación, utilizadas para crear las soluciones para las clases de GRAZ-01. . . . .	89
47. Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de forma, utilizadas para crear las soluciones para las clases de GRAZ-01. . . . .	90
48. Gráficas de la frecuencia de uso de las funciones y terminales para los mapas mentales, utilizadas para crear las soluciones para las clases de GRAZ-01. . . . .	90
49. Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de color, utilizadas para crear las soluciones para las clases de GRAZ-02. . . . .	91
50. Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de orientación, utilizadas para crear las soluciones para las clases de GRAZ-02. . . . .	92
51. Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de forma, utilizadas para crear las soluciones para las clases de GRAZ-02. . . . .	93
52. Gráficas de la frecuencia de uso de las funciones y terminales para los mapas mentales, utilizadas para crear las soluciones para las clases de GRAZ-02. . . . .	93
53. Ejemplo del procesamiento de una solución. Las imágenes representan el mapa de salida en cada una de las etapas del AVCMO. Esta solución en particular fue optimizada para clasificar la clase Persona de la base de datos GRAZ-02. . . . .	94



## Lista de tablas

Tabla		Página
1.	Ejemplos de máscaras usadas para aproximar la derivada de una imagen. . . . .	22
2.	Funciones y terminales para la construcción de los operadores visuales evolutivos <i>EVOs</i> . . . . .	66
3.	Parámetros utilizados en el proceso evolutivo. . . . .	70
4.	Rendimiento de los mejores individuos para todas las ejecuciones para la clase Bicicleta de GRAZ-01. . . . .	72
5.	Rendimiento de los mejores individuos para todas las ejecuciones para la clase Persona de GRAZ-01. . . . .	72
6.	Rendimiento de los mejores individuos para todas las ejecuciones para la clase Bicicleta de GRAZ-02. . . . .	73
7.	Rendimiento de los mejores individuos para todas las ejecuciones para la clase Persona de GRAZ-02. . . . .	73
8.	Rendimiento de los mejores individuos para todas las ejecuciones para la clase Carros de GRAZ-02. . . . .	74
9.	Comparación del rendimiento del modelo AVCMO en clasificar las imágenes de la base de datos de GRAZ. Los valores representan el porcentaje de clasificación para cada clase. . . . .	75
10.	Estructura de los operadores correspondientes a los mejores soluciones para cada una de las clases de la base de datos GRAZ. . . .	95

# Capítulo 1. Introducción

---

El sistema visual humano es capaz de localizar, identificar y categorizar objetos de manera eficiente. Sin importar las condiciones en las que se encuentren, ya sea que estén en diferentes posiciones, condiciones de iluminación, o se encuentren parcialmente ocluidos (Logothetis y Sheinberg, 1996). Si se quieren lograr sistemas con capacidades similares o superiores a aquellas de los seres vivos, es necesario entender el funcionamiento de los sistemas visuales naturales. De esta manera, emularlos resulta una estrategia atractiva para crear sistemas de visión artificial robustos. Von Neumann *et al.* (1966) fue el primero en demostrar que los sistemas artificiales basados en algoritmos computacionales pueden exhibir la característica de auto-replicación. Este comportamiento se puede reproducir desde el punto de vista de los algoritmos evolutivos, los cuales pueden ser aplicados en diversas áreas del conocimiento. Estos enfoques proponen resolver los problemas planteándolos en términos de problemas de optimización a fin de explotar el potencial del cómputo moderno.

La visión por computadora es un área de investigación multidisciplinaria, que busca construir sistemas artificiales para obtener información de imágenes digitales, transformándolas para generar representaciones numéricas de tal manera que puedan ser interpretadas por una computadora (Klette, 2014). Esta tarea implica abstraer información del mundo tal como: formas, colores, texturas, iluminación; y describirla de cierta manera que puedan ser utilizadas por un sistema computacional para tomar decisiones. Este proceso no es trivial, y a lo largo de la historia han surgido diversos enfoques para afrontarlo. El enfoque de esta tesis son las metodologías que se inspiran del funcionamiento del sistema visual humano, a fin de replicar su capacidad de percepción. El problema abordado en este trabajo es la clasificación de objetos, este consiste en la necesidad de encontrar un sistema capaz de asignar etiquetas semánticas a cada objeto. Es un problema fundamental para la visión por computadora, debido a que representa un paso necesario en tareas más complejas tales como la navegación, reconstrucción, seguridad, detección de fallos, por mencionar algunos (Deng *et al.*, 2014; Hernández *et al.*, 2015). De esta manera, en esta tesis se propone un nuevo modelo para la clasificación de objetos inspirado en el sistema de visión humano, y desarrollado a través de la evolución artificial.

Por otro lado, el espacio en que habitan los seres vivos, específicamente los animales terrestres, es un ambiente tridimensional el cual perciben a través de los sentidos. Así, la mayoría usa la visión como sensor principal para hacer la reconstrucción de su ambiente. El sentido de la vista se puede modelar como un sistema de información, el cual obtiene los datos por medio de los ojos. Se dice que estos proveen un conjunto de puntos, donde cada uno de estos puntos transmite al resto del sistema la cantidad de luz que percibe. Entonces, una imagen se representa por la cantidad de iluminación y reflectancia que ha sido proyectada en cierta parte de la retina (MacLeod y Boynton, 1979). Estos puntos forman una imagen, la cual, junto con un cúmulo de conocimiento previo, sirve al sistema para entender su ambiente (el propósito). De esta forma, la clasificación de objetos puede ser vista como el resultado de un proceso de interpretación que realiza el sistema de visión. Sin embargo, esto no se logra sólo con la información sensorial, la visión debe integrar el conocimiento suficiente para lograr una interpretación no-ambigua.

En la práctica, el proceso de visión se puede dividir en tres etapas, segmentación, reconstrucción y reconocimiento (Marr, 1982). La segmentación se encarga de resaltar la información importante en la imagen bidimensional; como son los cambio de intensidad, los contornos y la distribución y organización geométrica de los objetos (Perez y Olague, 2009). La reconstrucción se encargar de generar una representación tridimensional a partir de las imágenes bidimensionales. Por último, el reconocimiento se encarga de integrar el conocimiento previo a fin de buscar las características visuales conocidas en los objetos observados y poder conseguir el objetivo de identificarlos (Horaud y Monga, 1995). En este sentido, la clasificación de objetos está relacionado con dos de estas tareas principalmente: la localización, que está relacionada a la sementación, se refiere a identificar un objeto como una identidad independiente del entorno; y a la descripción, relacionada a la etapa de reconocimiento, consiste en encontrar características visuales del objeto que lo identifiquen, y además se puedan utilizar para agruparlo con otros objetos de características similares. Sin importar su tamaño, posición, orientación, condiciones de iluminación o si se encuentra dentro de un ambiente concurrido. Entonces, una parte importante de un sistema de clasificación es la capacidad de construir un descriptor del objeto a partir de un conjunto de propiedades visuales invariantes (Pinto *et al.*, 2008)

Actualmente, los sistemas de clasificación adoptan los siguientes enfoques: clasifi-

cación multi-clase, el cual busca definir una etiqueta al objeto de interés en base a un conjunto de etiquetas mutuamente excluyentes, tal es el caso del reto ILSVRC (Russakovsky *et al.*, 2015); o la clasificación binaria, que toma decisiones binarias para cada una de las etiquetas definidas, ejemplos de este tipo de sistemas son lo que abordan retos como la base de datos de Caltech (Fei-Fei *et al.*, 2007), el reto VOC (Everingham *et al.*, 2010) y GRAZ (Opelt *et al.*, 2006).

### 1.1. Atención visual

La atención visual es el proceso que establece una relación entre las distintas propiedades o características de una escena, las cuales son percibidas a través de un sistema visual, con el objetivo de seleccionar los aspectos más relevantes para la tarea que se está resolviendo (Treisman y Gelade, 1980; Dozal *et al.*, 2014).

Dentro del proceso de atención existen dos tipos de procesamiento de información que contribuyen al proceso de focalizar la atención en regiones específicas de la escena. El procesamiento *bottom-up* tiene que ver con estímulos no contemplados, es decir, información que conlleva a enfocar la atención a una región específica de la escena sin conocimiento *a priori*; este proceso está relacionado con la atención involuntaria.

Por otra parte, el procesamiento *top-down* es la atención voluntaria. En este caso, el procesamiento toma más tiempo y esfuerzo que en la atención involuntaria. Esto se debe a que el objetivo se encuentra inmerso entre distractores, y comparten algunas características. La atención voluntaria se lleva a cabo usando conocimiento *a priori* tal como, información contextual, expectativa u objetivos específicos. Es decir, existen algunos factores previamente contemplados para fijar la atención en características específicas (Dozal *et al.*, 2014). En la Figura 1 se muestra el concepto de atención *top-down*, donde el sistema se enfoca en los objetos de interés para la tarea en cuestión; es decir, se tiene un conocimiento previo sobre los objetos.

En este sentido, el modelo propuesto en este trabajo de tesis integra las tareas de localización de objetos, basado en el proceso de la atención visual, y la descripción de objetos, con el objetivo de crear un sistema de clasificación de objetos; partiendo de un enfoque de clasificación binaria en imágenes reales.

¿Donde está waldo?



¿Donde está la bicicleta?



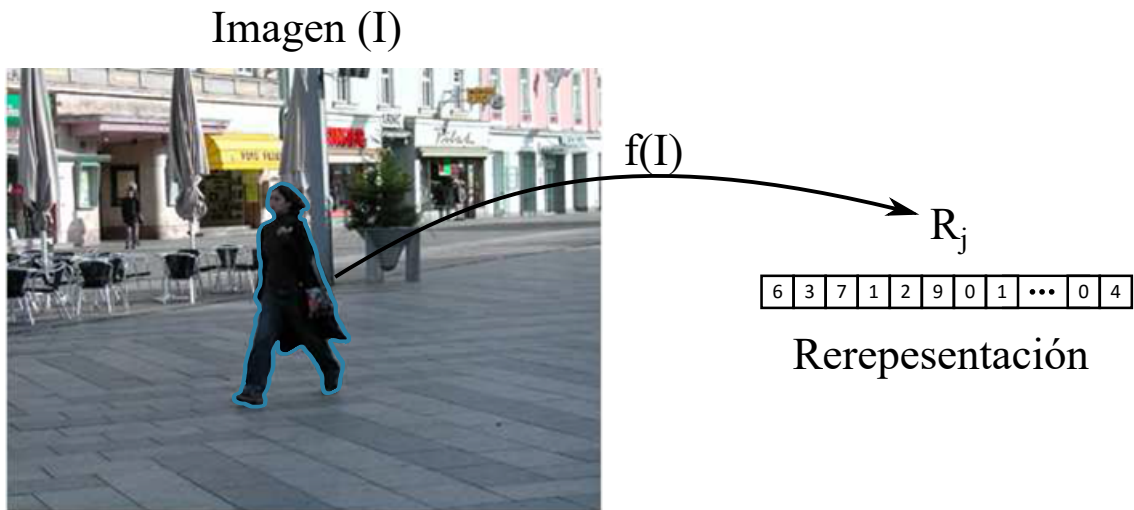
Figura 1: La atención visual se enfoca en los objetos de interés para la tarea que se está realizando.

## 1.2. Planteamiento del problema

En términos generales, el problema de clasificación de objetos consiste en identificar un objeto, o fenómeno desconocido, como miembro de una clase, o sub-categoría, de objetos (Clancey, 1984). Normalmente, las clases son estereotipos, definidos a través de características compartidas entre los elementos de la clase; además, las clases suelen estar organizadas dentro de una jerarquía. Así, el proceso de identificación consiste en emparentar observaciones de una entidad desconocida con características de las clases. Un ejemplo sencillo de este problema es la tarea de identificar una planta o animal, usando una guía de características, tales como color, estructura, tamaño.

Una característica esencial de un problema de clasificación es que el sistema que intenta resolverlo, conocido como clasificador, selecciona la clase a la que pertenece una observación partiendo de un conjunto de clases pre-definidas. Esto no significa que la respuesta correcta se encuentre dentro de este conjunto, pero un clasificador siempre buscará relacionar una observación con las clases conocidas. En el caso del problema de clasificación binaria se considera que existen dos sets de interés: el conjunto con la etiqueta de "clase", se asigna a las imágenes que contienen el tipo de objeto que se busca; el conjunto con la etiqueta de "no clase", utilizada cuando no se encuentra el objeto de interés.

Entonces, dada una imagen  $I$  y una clase de objetos  $C$ , el problema de presencia/ausencia consiste en determinar si  $I$  contiene algún elemento de  $C$ . Un ejemplo muy



**Figura 2:** La construcción de la representación  $R_j$ , también conocida como vector descriptor, debe relacionar la clase del objeto, sin importar las transformaciones visuales que pueda sufrir el objeto en la imagen  $I$ ; tales como traslación, rotación, escala o cambio de iluminación.

conocido de un sistema de clasificación es el dispositivo de reconocimiento de patrones descrito en el trabajo seminal de Holland (1992). Donde la imagen de entrada es procesada por un arreglo de sensores binarios de tamaño  $a \times b$ ; cuando se presenta una observación, cada sensor es ajustado a 1 si el patrón por reconocer se proyecta en el sensor, y a 0 en el caso contrario. Así, el número de posibles configuraciones para el arreglo de sensores es  $2^{ab}$ . Por lo tanto, la complejidad de dicho clasificador recae en el gran número de posibles configuraciones y la necesidad del sistema de encontrar la configuración correcta para reconocer un patrón dado.

De esta manera, resulta altamente complejo intentar clasificar punto por punto los elementos de la imagen de entrada. Una manera de reducir la complejidad del clasificador es definir un mapeo de la información visual contenida en la imagen, y establecer una representación generalizada de los objetos en la misma. Por lo tanto, dada una imagen  $I$  es necesario buscar una función que reduzca la información visual a una representación  $R_j$ ;  $f : I \rightarrow R_j$ , en base a las características visuales de los objetos en la escena; sin importar la posición, orientación, tamaño, color u alguna otra característica de los mismos, como lo muestra la Figura 2. Entonces, el problema de clasificación de objetos se puede reducir al problema de buscar dicha transformación; más aún, al establecer una métrica que mida la calidad de la clasificación, este problema se define como un proceso de optimización.

Por lo tanto, en este trabajo de tesis se aborda el problema de clasificación de objetos de la siguiente manera:

Dada una imagen de entrada  $I$  que pertenece a una base de datos de  $k$  clases de objetos, y un conjunto de etiquetas  $L = l_1, l_2, \dots, l_k$  para las clases en la base de datos; la tarea de clasificación de objetos puede expresarse como la búsqueda del mínimo argumento en la expresión:

$$Q = \operatorname{argmin}(c(f(I), L)) \quad (1)$$

donde  $c(\cdot)$  selecciona la etiqueta pertinente en base a la representación de la imagen brindada por  $f(I)$ , la cual devuelve una representación consistente del objeto  $R_j$  en la imagen  $I$ .

En la práctica, se han propuesto diferentes bases de imágenes para evaluar el desempeño de clasificación, con el objetivo de establecer un estándar de comparación del rendimiento en problemas reales de los diferentes clasificadores (Pinto *et al.*, 2008). Algunos ejemplos de estas bases de datos son: Caltech (Fei-Fei *et al.*, 2007), el reto VOC (Everingham *et al.*, 2010) y GRAZ (Opelt *et al.*, 2006). Estas bases de datos normalmente se construyen agrupando un gran número de imágenes en diferentes categorías, dependiendo de los objetos presentes en las mismas. Ejemplos de estas categorías son carros, personas, bicicletas, rostros, por mencionar algunos; además de incluir otro set de imágenes que no contienen ninguno de los objetos de las clases conocidas, este conjunto se conoce como no-clase o fondo.

De manera similar, se han propuesto diversas estrategias para medir el rendimiento de los sistemas de clasificación al procesar las imágenes de las bases mencionadas. Algunas métricas utilizadas por la comunidad son: área bajo la curva característica operativa del receptor (ROC, por sus siglas en inglés *receiver operating characteristic*), precisión promedio, razón de error equivalente (Clemente, 2015), por mencionar algunas.

En este sentido, la propuesta principal de este trabajo de tesis es un modelo bioinspirado para crear las representaciones de imágenes necesarias para un sistema de clasificación. La creación de dicha transformación se propone como un proceso de optimización, utilizando la base de imágenes de GRAZ (Opelt *et al.*, 2006) para compararse

con otros modelos del estado-del-arte propuestos por Opelt *et al.* (2006); Mutch y Lowe (2008); Huang *et al.* (2011); Ghodrati *et al.* (2012). El interés de comparar el sistema propuesto contra estos sistemas radica en que todos se enfocan en clasificar las bases de datos de GRAZ utilizando la razón de error equivalente para medir la calidad de los clasificadores. Además, los trabajos de Mutch y Lowe (2008); Huang *et al.* (2011); Ghodrati *et al.* (2012) proponen técnicas bio-inspiradas, similares a la propuesta en este trabajo de tesis.

### **1.3. Objetivo general**

Diseñar e implementar un modelo inspirado en el funcionamiento de las diferentes regiones de la corteza visual, que integre las tareas de detección y descripción de objetos por medio de la evolución artificial multi-objetivo, para resolver el problema de clasificación de objetos.

### **1.4. Objetivos específicos**

- Proponer un modelo para la clasificación de objetos basado en los modelos psicológicos, fisiológicos y computacionales de la corteza visual humano.
- Establecer un protocolo de descripción de la imagen inspirado en las etapas del procesamiento de la información visual en el sistema de visión natural.
- Extender el paradigma de la programación cerebral para optimizar los modelos usando un enfoque multi-objetivo.
- Integrar el paradigma de la atención visual como parte del proceso para crear descriptores de imágenes.
- Establecer criterios de evaluación estándares para medir el rendimiento de clasificación del modelo propuesto.
- Definir un protocolo de entrenamiento y pruebas utilizando una base de imágenes estándar para la comunidad de visión por computadora.
- Comparar el rendimiento de clasificación del modelo propuesto con respecto a otros modelos del estado-del-arte



## 1.5. Preguntas de investigación

Además de los objetivos ya descritos, esta tesis busca contestar las siguientes preguntas de investigación:

- ¿Cuales son los beneficios de aplicar el modelo de la atención visual al proceso de descripción de una imagen?
- ¿Qué etapa de tratamiento de la información visual es más apropiada para la construcción del vector descriptor?
- ¿Qué criterio es el más adecuado para medir la calidad del sistema para la detección de la región de la imagen ocupada por el objeto de interés?
- ¿Cual es la base de imágenes apropiada para comparar el modelo propuesto con otros del estado-del-arte?
- ¿Qué protocolo se debe seguir para evitar sobre entrenamiento en el proceso de optimización?

## 1.6. Alcance de la tesis

Una vez planteados los objetivos del trabajo de tesis, es necesario delimitar los alcances de este trabajo y definir las premisas. De esta manera, el modelo propuesto en esta tesis surge del estudio de diferentes modelos psicológicos y teorías neurofisiológicas de la corteza visual humana; así como de sistema computacionales diseñados para abordar problemas de atención visual, además de las tareas de descripción y reconocimiento de objetos. Con este fin, se establece lo siguiente:

- El modelo propuesto será utilizado en un esquema de clasificación binaria. Desde una perspectiva de presencia/ausencia del objeto de interés en una imagen.
- Se sigue el paradigma funcional descrito por Olague *et al.* (2014a). En el cual se busca caracterizar los objetos en base a un conjunto de transformaciones que ayuden encontrar información prominente sobre el objeto. Contrario a los métodos tradicionales, orientados a los datos, donde los objetos se caracterizan por características visuales específicas.

- La evaluación de la calidad del modelo de clasificación se hace utilizando las cinco clases que conforman los conjuntos GRAZ-01 y GRAZ-02 de la base de imágenes GRAZ (Opelt *et al.*, 2006). Por lo tanto, los clasificadores que se logren se enfocarán únicamente en la clasificación de imágenes de personas, bicicletas y carros.
- Dado el enfoque multi-objetivo que se toma en el proceso de optimización, no se puede establecer el concepto de una mejor solución. En el caso de los experimentos descritos en este trabajo, la mejor solución se refiere al individuo sintetizado, después de una ejecución del proceso evolutivo, que obtiene el mejor resultado al clasificar un conjunto de imágenes de prueba.

### 1.7. Organización de la tesis

Este documento se organiza de la siguiente manera:

- **Capítulo 1** Introducción.
- **Capítulo 2** Marco teórico. En el segundo capítulo se describen los aspectos teóricos utilizados para el desarrollo de este trabajo. Se describen los modelos base para el desarrollo del modelo propuesto; además de una introducción a las técnicas evolutivas y el enfoque multi-objetivo abordados para optimizar las soluciones.
- **Capítulo 3** Modelo AVCMO. Este capítulo está destinado a la descripción del modelo propuesto en este trabajo de tesis. Se detallan cada una de las etapas que lo componen, siguiendo la estructura jerárquica presente en los diferentes modelos de la corteza visual humana. También se describe la integración de las tareas de localización y descripción de objetos, las cuales realizan los dos algoritmos que sirvieron como pilares para el modelo propuesto en este trabajo.
- **Capítulo 4** Programación cerebral multi-objetivo. El cuarto capítulo se enfoca en explicar el funcionamiento del sistema evolutivo utilizado para el proceso de optimización de los operadores necesarios para el modelo de la AVCMO; presentando detalles de implementación del algoritmo evolutivo llamado programación cerebral.
- **Capítulo 5** Experimentos y resultados. El quinto capítulo presenta el ambiente de pruebas usado para mostrar la eficacia en la tarea de clasificación de objetos de los

modelos optimizados. Se hace una comparación con otros modelos del estado-del-arte y se muestran los resultados de un proceso de validación cruzada.

- **Capítulo 6** Conclusiones y trabajo futuro. Por último, se delinearán las conclusiones logradas después del desarrollo de esta tesis. Así como un conjunto de actividades que podrían realizarse como investigaciones futuras, partiendo del modelo y los resultados presentes en este documento.

## Capítulo 2. Marco teórico

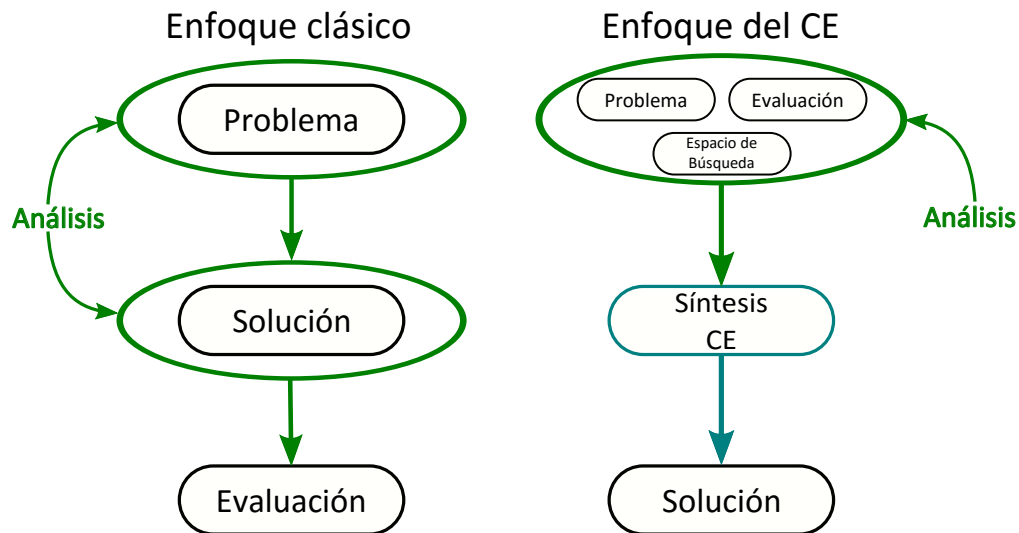
---

Numerosos sistemas naturales, tal como el cerebro, el sistema inmune o la sociedad; así como diversos sistema artificiales, como son las redes neuronales artificiales, el cómputo paralelo y distribuido y los sistemas evolutivos, por mencionar algunos, se caracterizan por los comportamientos que emergen de las interacciones, generalmente complejas, entre un gran número de componentes o sub-sistemas, los cuales normalmente siguen una estructura jerárquica (Chan, 2001). La dificultad para entender y diseñar este tipo de sistemas y poder resolver tareas complejas (i.e. en el caso de los sistemas naturales: ver, proteger al cuerpo de bacterias, control de población, y en los artificiales: optimización combinatoria, reconocimiento de patrones, distribución de recursos de procesamiento), recae en encontrar los componentes que mejor estructuren el sistema, y a la vez definir las interacciones óptimas para resolver la tarea en cuestión. Por esta razón, diversas comunidades científicas, como la visión por computadora, robótica, inteligencia artificial, biología, geología, por mencionar algunas, han enfocado su investigación en analizar y crear sistemas con estas características; sistemas formados por sub-sistemas independientes, y cuyo funcionamiento se basa en las interacciones de los mismos.

De esta manera, en este trabajo de tesis se plantea un sistema de clasificación de imágenes como un sistema complejo, compuesto de sub-sistemas independientes que colaboran para lograr una tarea en común. A continuación se detallan los diversos aspectos teóricos necesarios para el desarrollo de este proyecto. Empezando por describir el paradigma del cómputo evolutivo, seguido de las características de un proceso de optimización multi-objetivo. Posteriormente, se detalla el enfoque funcional abordado por el modelo propuesto en esta tesis; presentado el concepto de imagen como el gráfico de una función, al mismo tiempo que la definición de las diferentes operaciones sobre las imágenes que sirven como base para la creación de las operaciones visuales.

### 2.1. La evolución como paradigma computacional

En el trabajo seminal de Charles Darwin, "El origen de las especies"(Darwin y Bynum, 1859), se fundamenta la teoría de la evolución biológica, la cual establece que todos los seres vivos que habitan la tierra son descendientes directos de seres que habitaron el planeta hace mucho tiempo. Esta teoría busca describir la capacidad de adaptación y diseño



**Figura 3: Descripción general del funcionamiento de dos enfoques para resolver problemas.**

que exhiben los diversos sistemas biológicos. De esta manera, la teoría de la evolución de los sistemas naturales resulta de particular interés en las ciencias que se inspiran de la naturaleza para la resolución de problemas. En el caso de esta tesis, el enfoque principal se encuentra en las ciencias computacionales que se basan en algoritmos que buscan abstraer y reproducir las capacidades de adaptación y diseño del proceso de evolución biológica.

El *cómputo evolutivo* (CE) es un área de la inteligencia artificial que se inspira en el proceso de la evolución biológica y la teoría de la selección natural. Además, los trabajos por Eigen (1973), Fogel *et al.* (1966), Holland (1975) y Koza (1992) establecen las bases teóricas de las diferentes técnicas utilizadas en esta área de investigación (Eiben y Smith, 2003).

A diferencia de los paradigmas tradicionales para resolver problemas, donde se realiza un análisis profundo del problema y se busca establecer una solución analítica para resolverlo, el paradigma del CE se enfoca en optimizar soluciones a través de un proceso de búsqueda. Cabe notar que también es necesario realizar un análisis profundo del problema, con el objetivo de establecer un espacio de búsqueda factible que brinde soluciones al problema en cuestión. La Figura 3 muestra un esquema de las diferencias entre el enfoque del CE y técnicas tradicionales.

El primer diagrama muestra un enfoque clásico, donde se busca analíticamente las

condiciones del problema. Por ende, es necesario realizar un análisis minucioso y revisar diferentes perspectivas para lograr soluciones diferentes. Además, cuando se tienen soluciones diferentes es necesario establecer un criterio circunstancial para definir qué solución es la mejor. Por su parte, el enfoque del CE explota un criterio o función de evaluación para probar un gran número de posibles soluciones, y así encontrar la mejor a partir de un proceso de búsqueda. Esta función de evaluación se utiliza como guía para el proceso evolutivo y se conoce como función de aptitud (*fitness function*).

A través de esta función, el CE permite establecer los problemas como problemas de optimización. Entonces, al emplear un enfoque evolutivo, no es necesario definir una solución específica al problema en cuestión. Sino establecer un espacio de las posibles soluciones y una función que dirija el proceso hacia una solución óptima.

De esta manera, en el CE existen diferentes técnicas o algoritmos evolutivos (AE), los cuales comparten muchas características, pero varían detalles como la representación de las soluciones, las técnicas que se utilizan para encontrar nuevas soluciones y algunos parámetros del proceso evolutivo. Sin embargo, todos funcionan bajo el mismo principio; es decir, son heurísticas estocásticas inspiradas en la teoría de la evolución biológica y cuyo funcionamiento se basa en los siguientes mecanismos (Eiben y Smith, 2003):

1. El genotipo, la codificación que representa una posible solución, a cada solución se le conoce como *individuo*. Un conjunto de individuos se conoce como *población*.
2. La función objetivo, es una función que evalúa el rendimiento de un individuo al resolver el problema en cuestión. El aplicar un individuo para resolver un problema se le conoce como *fenotipo*. Esta función asigna una *aptitud* a cada solución, esta representa la calidad de adaptación de un individuo a un *ambiente*, el cual representa el problema.
3. Selección, es una estrategia para elegir un conjunto de individuos, los cuales se utilizan como base para generar nuevas soluciones.
4. Mecanismos de variación, son técnicas para generar nuevas soluciones (*hijos*), a partir de individuos previamente seleccionados. Existen dos mecanismos principales: la *cruza o recombinación*, que consiste en utilizar la información de dos o más

individuos para generar una nueva solución; y la *mutación*, la cual toma un individuo y utiliza un proceso estocástico para generar una nueva.

5. Mecanismo de supervivencia, es una metodología de selección para decidir qué individuos de la población seguirán en la siguiente iteración del proceso evolutivo, cada iteración se conoce como *generación*.

Como se mencionó previamente, existen diferentes tipos de AE. Sin embargo, la técnica utilizada en este trabajo de tesis se inspira principalmente de la *programación genética* propuesta por John Koza (1992).

### 2.1.1. Programación genética

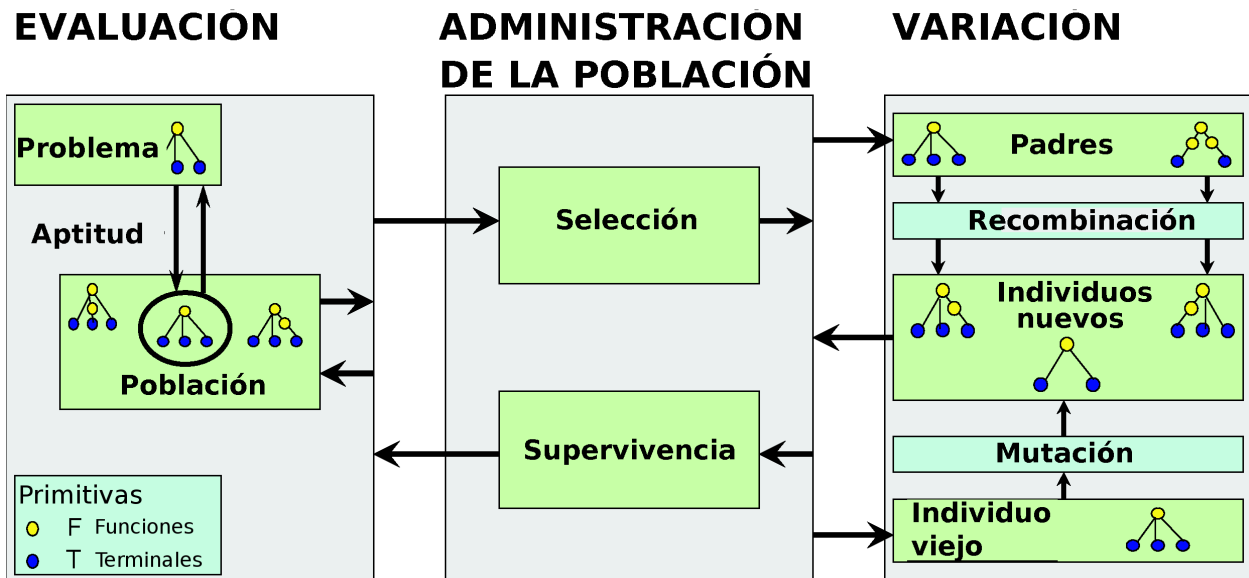
La programación genética (GP<sup>1</sup> por sus siglas en inglés *genetic programming*) es una técnica evolutiva que permite resolver problemas de forma automática (Poli *et al.*, 2008); a través de la síntesis de modelos computacionales por medio proceso de optimización.

La GP es similar a las otras técnicas evolutivas, la diferencia principal radica en la representación de los individuos. En la GP las soluciones se definen como árboles, con una estructura similar a la utilizada en lenguajes de alto nivel como LISP. A diferencia de los algoritmos genéticos, que se utilizan generalmente para la optimización de parámetros, la GP se enfoca en la síntesis de programas o funciones computacionales. Por ende, la GP se denota como una técnica para generar programas de manera automática. En las palabras de Koza "La programación de computadoras por medio de la selección natural"(Koza, 1992).

El algoritmo canónico de GP construye soluciones utilizando elementos de un conjunto de funciones  $F$  en los nodos, o ramas del árbol; las cuales usan como variables de entrada elementos de un conjunto de terminales  $T$ . Los elementos del conjunto  $F \cup T$  se denominan *primitivas* del algoritmo de GP. La selección de estas primitivas depende del dominio del problema, y son estas las que definen el espacio de soluciones. Además, las operaciones genéticas, cruza y mutación, son más complejas comparadas con las

---

<sup>1</sup>La abreviación **GP** puede referirse a dos cosas: cuando se hace referencia a la metodología como parte del computo evolutivo, o la implementación de un sistema de este tipo para resolver un problema en particular.



**Figura 4: Estructura básica de un algoritmo de GP. Los individuos representados como árboles, construidos con las primitivas; además, se muestran los mecanismos de variación para la creación de nuevos individuos. Diagrama propuesto originalmente por Trujillo y Olague (2008).**

operaciones usadas en los algoritmos genéticos; esto se debe a la representación de los individuos. La Figura 6 presenta un diagrama a bloques del algoritmo canónico de la GP.

En un algoritmo de GP, el objetivo de los operadores de variación es generar individuos más aptos. La operación de cruce busca hacerlo al combinar información de dos individuos y así crear mejores soluciones; cabe notar que todos los individuos de la población pueden ser seleccionados como padres. Sin embargo, los individuos más aptos tienen mayor probabilidad de ser seleccionados. Por su parte, la operación de mutación crea una nueva solución cambiando, de manera aleatoria, porciones de un individuo seleccionado. Además, esta operación establece un mecanismo para mantener la diversidad en la población, al introducir soluciones distintas a las existentes (Trujillo, 2008).

Así mismo, el otro aspecto importante para un algoritmo de GP es la función objetivo o función de aptitud. Esta establece una métrica para evaluar la calidad de cada individuo para resolver el problema en cuestión. En el caso de esta tesis se plantea utilizar dos funciones para medir la eficacia de los modelos, ya que se busca evaluar su calidad al resolver dos tareas distintas. Por lo tanto, es necesario aplicar un enfoque de optimización multi-objetivo.



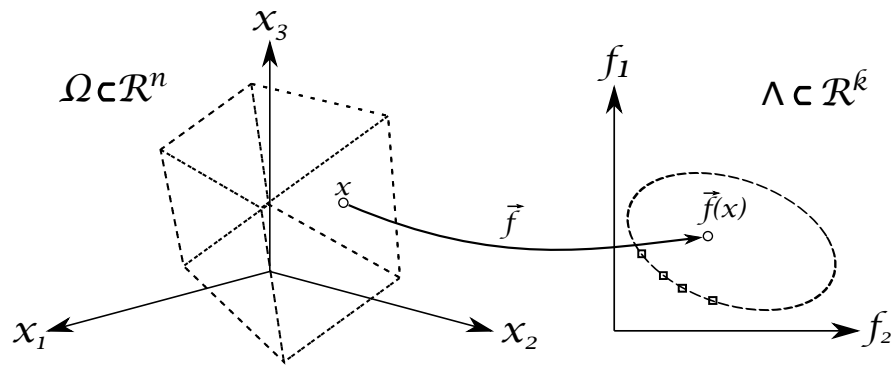


Figura 5: Espacio de decisión que se relaciona al espacio de objetivos por medio del mapeo  $\vec{f}$ .

### 2.1.2. Evolución multi-objetivo

Establecer la optimalidad de una solución resulta trivial en un enfoque mono-objetivo. Sin embargo, existen problemas que requieren evaluar las soluciones utilizando varios objetivos. Además, en muchas ocasiones estos objetivos no son conmensurables o están en conflicto. Un problema de optimización multi-objetivo puede ser descrito como, la tarea de encontrar un vector de variables de decisión que cumpla con un conjunto de restricciones y optimice una función vectorial donde cada elemento representa una función objetivo. Una manera de evaluar una solución en un ambiente multi-objetivo es fijar una jerarquía entre los criterios considerados. Dicha jerarquía determina las propiedades de la solución deseada y puede ser integrada antes o después del proceso de optimización. Considerar la jerarquía antes del proceso de optimización implica especificar el impacto que tiene cada criterio en la evaluación de la solución, de tal manera que el proceso está sesgado a encontrar una solución apegada a dicha jerarquía. Ejemplos de este enfoque son la agregación aditiva, multiplicación de objetivos, combinaciones no lineales de objetivos, ordenamiento lexicográfico, por mencionar algunos (Dunn Rivera, 2006).

El enfoque contrario a esto es la integración *a posteriori* de la jerarquía. En este caso, el proceso de optimización no se limita por suposiciones sobre el impacto de cada criterio. Así, se requiere un criterio externo para la selección de la solución final. En esta tesis, se considera la optimización multi-objetivo (MO) con una integración *a posteriori* de las preferencias en los criterios de evaluación.

### 2.1.2.1. Optimización multi-objetivo

El pionero en tratar problemas de optimización MO fue Pareto, presentado las bases para la optimización en áreas como la teoría económica y la investigación de operaciones (Pareto, 1897). Sin embargo, fueron Kuhn, Tucker y Hurwicz quienes formalizan los conceptos de la optimización MO que se utiliza actualmente (Trujillo y Olague, 2008).

Un problema MO requiere del estudio de dos espacios complementarios, el espacio de las variables de decisión y el espacio de los objetivos (espacio de aptitud multi-dimensional) como lo muestra la Figura 5. En el caso de las funciones reales, los espacios se relacionan a través de un mapeo  $\vec{f} : \mathbb{R}^n \rightarrow \mathbb{R}^k$ ; mientras que las restricciones sobre  $x$  definen una región  $\Omega \subset \mathbb{R}^n$  en el espacio de decisión y una región  $\Lambda \subset \mathbb{R}^k$  sobre el espacio de las funciones objetivo. Bajo estas condiciones, las soluciones óptimas se encuentran en la frontera de  $\Lambda$  y se conocen como *frente Pareto* (Dunn Rivera, 2006). El concepto de optimalidad en el sentido de Pareto se define a través de las siguientes relaciones:

- **Relación de dominancia de Pareto:** Dados  $k$  objetivos y un conjunto ordenado  $N = \{1, \dots, k\}$ , un vector de objetivos  $\vec{f}^u$  domina a otro vector  $\vec{f}^v$ ,  $(\vec{f}^u \preceq \vec{f}^v) \Leftrightarrow \forall i \in N, f_i^u \leq f_i^v \wedge \exists j \in N | f_j^u < f_j^v$ .
- **Optimalidad Pareto:** Un vector solución  $x^* \in \Omega$  es óptimo en el sentido de Pareto si  $\forall x \in \Omega$  se cumple que  $\forall i \in N, f_i(x^*) = f_i(x) \vee \exists j \in N | f_j(x^*) < f_j(x)$ .
- **Conjunto óptimo Pareto:** Para un problema multi-objetivo  $\vec{f}(x)$ , el conjunto de soluciones óptimas en el sentido de Pareto se define como:

$$\mathcal{P}^* = \{x \in \Omega | \nexists x' \in \Omega \setminus \{x\} | \vec{f}(x') \preceq \vec{f}(x)\} \quad (2)$$

- **Frente Pareto:** Para un problema multi-objetivo con un vector de objetivos  $\vec{f}(x)$  y un conjunto óptimo Pareto  $\mathcal{P}^*$ , el frente Pareto está dado por:

$$\mathcal{PF}^* = \{\mathbf{u} = (f_1(x), \dots, f_k(x)) | \mathbf{x} \in \mathcal{P}^*\} \quad (3)$$

Partiendo de estas definiciones, encontramos que si los objetivos de un problema MO están en conflicto no existe una solución óptima única, existen varias soluciones óptimas y se dice que son óptimas en el sentido de Pareto. Además, la definición previa establece las condiciones de un *Frente Pareto Verdadero*, pero debido a las heurísticas utilizadas en esta tesis los resultados que se muestran son los mejores frentes no dominados que se logran después del proceso de búsqueda; que en caso de ser óptimos, deberían ser iguales al Frente Pareto Verdadero. No obstante, a cada instancia encontrada se le denominará *Frente Pareto*, ya que no se demuestra que dicho frente corresponda o pertenezca al Frente Pareto Verdadero.

### 2.1.2.2. Algoritmos evolutivos multi-objetivo

Los algoritmos evolutivos multi-objetivo (*Multiobjective Evolutionary Algorithm: MOEA*) son muy similares a un algoritmo evolutivo convencional. La diferencia principal se encuentra en los proceso de selección y supervivencia. La evaluación de la calidad de cada individuo debe considerar la naturaleza MO del problema, y así guiar el proceso de optimización hacia la construcción de un frente Pareto. Para esto, los MOEA utilizan criterios de dominancia Pareto. Por otra parte, como el problema no tiene una solución única, es importante buscar soluciones distribuidas a lo largo de todo el frente. Para lograr esto, un MOEA debe promover la diversidad de la población en el espacio de objetivos. Esto se logra utilizando métodos basados en núcleos, métodos de clustering, métodos de histogramas, o utilizando el concepto de dominancia- $\epsilon$  (Trujillo y Olague, 2008).

Los ejemplos más comunes de MOEAs son: el propuesto por Deb *et al.* (2000) conocido como algoritmo genético de ordenamiento no dominado (*Non-Dominated Sorting Genetic Algorithm NSGA-II*), el método de selección de Pareto basada en cobertura (*Pareto Envelope Based Selection PESA*) de Corne *et al.* (2000) y el utilizado para el proceso evolutivo de esta tesis, el algoritmo evolutivo de fuerza de Pareto (*Improved Strength Pareto Evolutionary Algorithm, SPEA2*) propuesto por Zitzler *et al.* (2001).

## 2.2. Operaciones sobre imágenes

En busca de establecer un enfoque funcional, en contraste al enfoque basado en los datos que siguen otros modelos, en este trabajo se sigue la hipótesis que la información

visual en una imagen, se puede expresar como la funcional de una escena y que todas las características que la describen son intrínsecas a dicha funcional. De esta manera, la descripción de una imagen se puede hacer en base a la respuesta de una composición de funciones (Clemente, 2015). Por lo tanto, una imagen digital se define como la representación de esta funcional, es decir como el gráfico de una función.

**Definition 1** *Imagen como el gráfico de una función.* Sea  $f$  una función  $f : U \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ . El gráfico, o imagen,  $I$  de  $f$  es el subconjunto de  $\mathbb{R}^3$  que consiste de los puntos  $(x, y, f(x, y))$ , donde el par ordenado  $(x, y)$  es la coordenada en  $U$  y  $f(x, y)$  es el valor de la imagen en dicho punto. Así, la imagen  $I = \{(x, y, f(x, y)) \in \mathbb{R}^3 | (x, y) \in U\}$

Por ende, el flujo de información visual dentro de modelo se puede definir a través de una composición de funciones, como una serie de transformaciones u operadores sobre la superficie descrita por la imagen.

**Definition 2** *Operador visual.* Un operador visual (OV) es una transformación que se aplica a la imagen  $I_q$ , en busca de extraer información prominente de la misma, tales como: bordes, esquinas, contornos, forma, texturas por mencionar algunas. La salida del OV se conoce como mapa visual.

A continuación se detallan las operaciones implementadas en este trabajo de tesis.

### 2.2.1. Convolución

La convolución es una operación básica en el procesamiento de imágenes (Pajares M y De La Cruz Garcia, 2008). Empezando por el caso de una señal de una dimensión, la convolución entre dos funciones  $f(x)$  y  $g(x)$ , se escribe como  $f(x) * g(x)$ , definida en el continuo de la siguiente manera:

$$f(x) * g(x) = \int_{-\infty}^{\infty} f(\alpha)g(x - \alpha)d\alpha \quad (4)$$

donde  $\alpha$  es la variable de integración que define el valor de desplazamiento de la operación de convolución.

En base a la teoría aplicada al tratamiento digital de señales, el filtrado de las mismas se puede hacer ya sea en el dominio espacial como en el dominio de la frecuencia. En este último, la operación de filtrado se reduce a una multiplicación de las transformadas de las señales; mientras que en el dominio espacial, el proceso de filtrado se efectúa en base a una convolución. De esta manera, dadas dos funciones imagen  $f$  y  $h$  con transformadas de Fourier  $F$  y  $H$ , el teorema de convolución se define como:

$$[f * h][(x, y)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(m, n)h(x - m, y - n)dmdn \quad (5)$$

Usando la transformada de Fourier, este proceso se define como sigue:

$$F_0(u, v) = \mathfrak{F}\{f * h(x, y)\} = \mathfrak{F}\{f(x, y)\}\mathfrak{F}\{h(x, y)\} = F(u, v)H(u, v) \quad (6)$$

la función  $H(u, v)$  define un filtro espacial lineal, y se conoce como la *función de transferencia* del filtro. El producto  $F_0(u, v) = F(u, v)H(u, v)$  es la transformada de Fourier de la salida del filtro. Por lo tanto, la función imagen resultante del filtrado  $f_0(x, y)$  es la transformada inversa de Fourier de  $F_0$ . Así, del teorema de convolución se obtiene,

$$f_0(x, y) = \mathfrak{F}^{-1}\{F(u, v)H(u, v)\} = f * h[(x, y)]; h(x, y) = \mathfrak{F}^{-1}\{H(u, v)\} \quad (7)$$

la función  $h(x, y)$  se denomina como *respuesta impulso* del filtro. Por consiguiente, partiendo del teorema de convolución, un filtro se puede definir por la función de transferencia  $H$  o por la respuesta impulso  $h$ .

Las definiciones previamente dadas se encuentran en el dominio continuo, pero debido a que en esta tesis se trata con imágenes digitales, el teorema de la convolución en el dominio discreto se expresa reemplazando la integral por una sumatoria, por lo tanto:

$$f_0(x, y) = f * h = \sum_m \sum_n f(m, n)h(x - m, y - n) = \sum_m \sum_n f(x - m)(y - n)h(m, n) \quad (8)$$

### 2.2.2. Derivada de una imagen

Como se mencionó previamente, las imágenes digitales son funciones discretas, por lo tanto la definición matemática de una derivada no se puede aplicar como tal. Consecuentemente, existen diversos algoritmos en la literatura que aproximan la derivada de una imagen continua  $f(x, y)$ . Una técnica común para la diferenciación de una imagen es utilizar el concepto del gradiente (Pajares M y De La Cruz Garcia, 2008). Entonces, el gradiente de una imagen  $f(x, y)$  en el punto  $(x, y)$  es el vector:

$$\nabla f(x, y) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\delta f(x, y)}{\delta x} \\ \frac{\delta f(x, y)}{\delta y} \end{bmatrix} \quad (9)$$

Así, el gradiente indica la dirección de la variación máxima en  $f(x, y)$ ; mientras que el módulo de  $\nabla f(x, y)$  es la magnitud de dicha variación,

$$\|\nabla f(x, y)\| = \left[ \left( \frac{\delta f(x, y)}{\delta x} \right)^2 + \left( \frac{\delta f(x, y)}{\delta y} \right)^2 \right]^{\frac{1}{2}} \quad (10)$$

en la práctica, el módulo se aproxima con la siguiente expresión:

$$\|\nabla f(x, y)\| \approx |G_x| + |G_y|, \quad (11)$$

mientras que la dirección del gradiente está dada por:

$$\alpha(x, y) = \arctan \left( \frac{G_y}{G_x} \right) \quad (12)$$

Estas expresiones se extienden a una imagen digital de la siguiente manera:

$$\|I(u, v)\| = [(h_u * I(u, v))^2 + (h_v * I(u, v))^2]^{\frac{1}{2}} \quad (13)$$

$$\alpha(u, v) = \arctan \left( \frac{h_v * I(u, v)}{h_u * I(u, v)} \right) \quad (14)$$

donde  $h_u$  y  $h_v$  son máscaras, o núcleos, que operan sobre la imagen en la dirección de  $u$  y  $v$  respectivamente.

**Tabla 1: Ejemplos de máscaras usadas para aproximar la derivada de una imagen.**

Roberts	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$
Prewit	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$
Sobel	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$

Existen varias máscaras que usando la operación convolución aproximan la derivada de la imagen por diferencias finitas; algunas de estas máscaras son la propuestas por Roberts, Prewit y Sobel, presentas en la Tabla 1

Una característica particular del filtro Sobel es que además de aproximar la derivada, tiene un efecto de suavizado sobre la imagen. Debido a que la operación de diferenciación realza el ruido en la imagen, el efecto de suavizado es una característica atractiva. Marr y Hildreth (1980) propusieron utilizar un filtro Gaussiano para calcular el Laplaciano como sigue:

$$h(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (15)$$

$$\nabla^2 h = \left( \frac{r^2 - \sigma^2}{\sigma^4} \right) e^{-\frac{r^2}{2\sigma^2}} \quad (16)$$

En la actualidad, este tipo de filtros son muy populares, debido a su robustez al ruido y eficacia en calcular la primera y segunda derivada y el Laplaciano. Deriche (1993) propone utilizar un filtro recursivo para calcular la derivada de una imagen, aplicando la convolución de la primera derivada de la función Gaussiana y la imagen. En el presente trabajo se optó por utilizar una máscara Gaussiana para el calculo de la primera y segunda derivadas de las imágenes digitales.

### 2.2.3. Suavizado

La operación de suavizado en una imagen digital consiste en asignar el valor a cada pixel a partir de la información de su vecindario, es decir, los pixeles a su alrededor (ver Figura 6). Uno de los algoritmos más comunes para realizar un suavizado, gracias a

1	2	3
4	(u,v)	5
6	7	8

**Figura 6: Vecindario de tamaño  $3 \times 3$  alrededor del pixel  $(u, v)$ .**

su fácil implementación, es aplicar un promedio del vecindario. De la siguiente manera, dada una imagen  $I(u, v)$ , una imagen suavizada  $I_s(u, v)$ , donde el valor de cada punto  $(u, v)$  se define por el promedio de la intensidad de los pixeles de  $F$  que forman parte de un vecindario predefinido  $S$ . Esta operación se define como:

$$I_s(u, v) = \frac{1}{|S|} \sum_{(m,n \in S)} I(m, n) \quad (17)$$

para todos los pixeles  $(u, v)$  en  $I$ ; donde  $S$  es un conjunto de coordenadas de los puntos del vecindario de  $(u, v)$  y  $|S|$  es el tamaño del vecindario (Pajares M y De La Cruz Garcia, 2008).

Otra estrategia para realizar esta operación de vecindario es por medio de un núcleo Gaussiano de dos dimensiones, con media 0 y desviación estándar  $\sigma$ . Esta operación se conoce como suavizado Gaussiano, donde la máscara que se utiliza para el proceso de convolución se define como:

$$G(i, j) = e^{-\frac{i^2+j^2}{2\sigma^2}} \quad (18)$$

La Figura 7 presenta un ejemplo de este proceso usando  $\sigma = 2$ . Cabe notar que existe un gran número de algoritmos para lograr el suavizado de una señal, tales como el suavizado aditivo, exponencial, por kernels, Laplaciano, spline, por mencionar algunos; sin embargo en este trabajo de tesis se optó por aplicar el suavizado Gaussiano, dado su utilidad para resaltar las estructuras de los objetos en diferentes escalas.



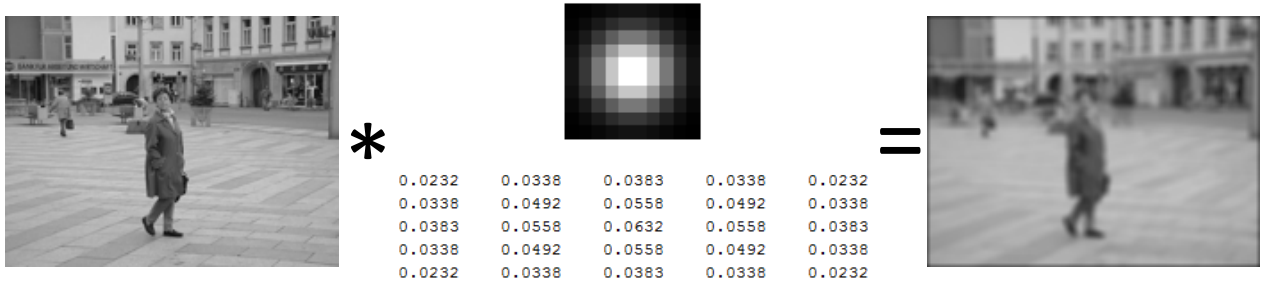


Figura 7: Ejemplo de suavizado Gaussiano con  $\sigma = 2$  y un núcleo de  $5 \times 5$ .

#### 2.2.4. Operación de umbral

La operación de umbral sirve para convertir una imagen en escala de grises a una imagen binaria. Este proceso se realiza aplicando el método propuesto por Otsu (1975) para establecer el valor de umbral. Este algoritmo supone que la imagen contiene dos clases de píxeles que siguen un histograma bi-modal; para esto, se calcula una frontera de decisión para separar las dos clases, el cual busca un valor de umbral que minimiza la varianza entre los elementos de la misma clase; definida como la suma ponderada de las varianzas de las dos clases:

$$\sigma_{\omega}^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) \quad (19)$$

los pesos  $\omega_0$  y  $\omega_1$  son las probabilidades de las dos clases al separarlas por el umbral ( $t$ ) y  $\sigma_{0,1}$  son las varianzas de las clases. Las probabilidades  $\omega_{0,1}(t)$  se calculan en base a un histograma, contando los elementos de cada uno de los valores de intensidad:

$$\omega_0(t) = \sum_{i=0}^{t-1} p(i) \quad (20)$$

$$\omega_1(t) = \sum_{i=t}^{L-1} p(i) \quad (21)$$

En su trabajo, Otsu demostró que minimizar la varianza intra clase es equivalente que maximizar la varianza entre las clases, es decir:

$$\sigma_b^2(t) = \sigma^2 - \sigma_{\omega}^2(t) = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 = \omega_0(t)\omega_1(t)[\mu_0(t) - \mu_1(t)]^2 \quad (22)$$



**Figura 8: Ejemplo del uso de la operación de umbral de Otsu para transformar una imagen en escala de grises a una imagen binaria.**

descrito en términos de las probabilidades  $\omega$  y medias  $\mu$  de dos clases; donde las medias se calculan de la siguiente manera:

$$\mu_0(t) = \sum_{i=0}^{t-1} ip(i)/\omega_0 \quad \mu_1(t) = \sum_{i=t}^{L-1} ip(i)/\omega_1 \quad \mu_T(t) = \sum_{i=0}^{L-1} ip(i)$$

entonces, las probabilidades y las medias se pueden calcular de forma iterativa, lo que resulta en un algoritmo fácil de implementar. La Figura 8 muestra un ejemplo del uso de esta operación. Cabe notar que existen diversos algoritmos para calcular el valor del umbral, tales como los métodos basados en histogramas, basados en entropía, basados en atributos de los objetos, así como métodos locales. En este trabajo se optó por el método de Otsu dado que logra buenos resultados y su facilidad de implementación.

### 2.2.5. Operaciones punto por punto

Además de las funciones ya descritas, en este trabajo se implementaron otras operaciones para el tratamiento de imágenes. Sean dos imágenes de entrada  $A(u, v)$  y  $B(u, v)$ , y la imagen de salida  $C(u, v)$  del mismo tamaño, se presentan las siguientes operaciones:

- **Operaciones aritméticas:** suma, resta, multiplicación, división, potenciación.

$$C(i, j) = A(i, j) + B(i, j) \quad C(i, j) = A(i, j) - B(i, j) \quad C(i, j) = A(i, j) \times B(i, j)$$

$$C(i, j) = A(i, j)/B(i, j) \quad C(i, j) = [A(i, j)]^2 \quad C(i, j) = \sqrt{A(i, j)}$$

- **Operaciones con un escalar  $k$ :** suma, resta, multiplicación, división.

$$C(i, j) = A(i, j) + k \quad C(i, j) = A(i, j) - k \quad C(i, j) = k \times A(i, j)$$

$$C(i, j) = A(i, j)/k \text{ donde } k \neq 0$$

- **Valor absoluto:** valor absoluto, suma absoluta y resta absoluta.

$$C(i, j) = |A(i, j)| \quad C(i, j) = |A(i, j) + B(i, j)| \quad C(i, j) = |A(i, j) - B(i, j)|$$

- **Operaciones de redondeo:** redondeo, techo, piso.

$$C(i, j) = \text{round}(A(i, j)) \quad C(i, j) = \lceil A(i, j) \rceil \quad C(i, j) = \lfloor A(i, j) \rfloor$$

- **Infimo y supremo.**

$$C(i, j) = \min(A(i, j), B(i, j)) \quad C(i, j) = \max(A(i, j), B(i, j))$$

- **Exponencial y logaritmo.**

$$C(i, j) = e^{A(i, j)} \quad C(i, j) = \ln(|A(i, j)|) \text{ si } A(i, j) \neq 0$$

- **Complemento:** la imagen de entrada se normaliza y se calcula el complemento

$$C(i, j) = |A(i, j) - 1|$$

### 2.2.6. Morfología

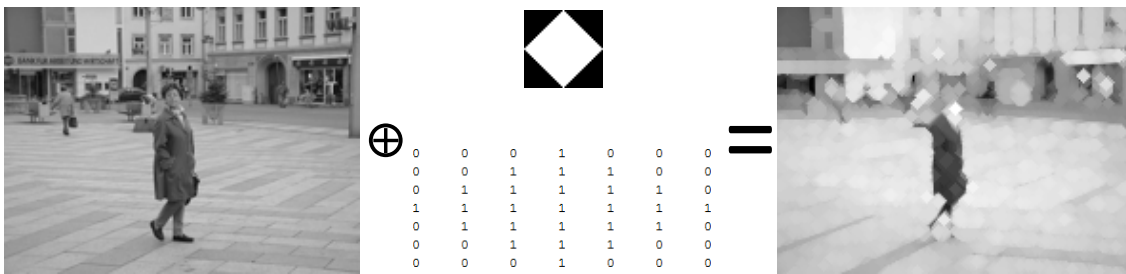
Otro conjunto de operaciones que se implementaron para construir las soluciones propuestas en esta tesis son operaciones de morfología matemática, esto con el objetivo de extraer información de las imágenes sobre la forma de los objetos. El objetivo de la morfología matemática es analizar estructuras geométricas, partiendo de conceptos topológicos y geométricos tal como tamaño, forma, convexidad, conectividad y distancia geodésica. La morfología matemática simplifica las imágenes y preservan las formas principales de los objetos (Pajares M y De La Cruz Garcia, 2008). Las operaciones básicas de la morfología para el procesamiento de imágenes son erosión, dilatación, apertura y cerradura.

#### Dilatación y erosión

La operación morfológica de la dilatación  $\oplus$  combina dos conjuntos usando la adición de vectores, conocida como adición de conjuntos de Minkowski. La dilatación de dos conjuntos de puntos  $A$  y  $B$ , donde sus elementos son puntos en un espacio Euclídeo  $E^n$ ,  $A \otimes B$  es el conjunto puntos de todas las posibles adiciones vectoriales entre los elementos de  $A$  y los elementos de  $B$ .

$$A \oplus B = \{c \in E^n : c = a + b \forall a \in A, b \in B\} \quad (23)$$

Esta definición se extiende a imágenes en escala de grises de la siguiente manera.



**Figura 9: Ejemplo del uso de la operación morfológica de dilatación usando un elemento estructural en forma de diamante.**

Sea una imagen de entrada  $I(u, v)$  de tamaño  $M \times N$  y un elemento estructural  $b(i, j)$ , similar a las máscaras descritas durante las operaciones de filtrado, una sub-imagen de tamaño  $m \times n$ . La operación de dilatación se define como:

$$(I \oplus b)(u, v) = \max_{\substack{0 \leq i \leq m-1 \\ 0 \leq j \leq n-1}} \{a(u-i, v-j) + b(i, j)\} \quad (24)$$

La Figura 9 muestra un ejemplo de esta operación partiendo de un elemento estructural en forma de diamante; en este trabajo también se utilizan elementos estructurales en forma de cuadrado y disco.

La operación morfológica de la erosión  $\ominus$  (notada como  $\otimes$  por algunos autores) combina dos conjuntos, similar a la dilatación, pero esta operación lo hace a través de la substracción de vectores. Es la operación dual de la dilatación; que no debe ser confundido con el concepto de inversa, dado que ni la erosión ni la dilatación son transformaciones invertibles. La erosión se define como:

$$A \ominus B = \{c \in E^n : c + b \in A \forall b \in B\} \quad (25)$$

es decir, los elementos del conjunto de salida son aquellos elementos de  $A$  cuya suma con todos los posibles elementos de  $B$  existen en  $A$ . Este concepto se extiende al tratamiento de imágenes de gris de la siguiente forma:

$$(I \ominus b)(u, v) = \min_{\substack{0 \leq i \leq m-1 \\ 0 \leq j \leq n-1}} \{I(u+i, v+j) - b(i, j)\} \quad (26)$$

donde la operación elige el mínimo del valor de la operación  $I - b$  en un vecindario definido

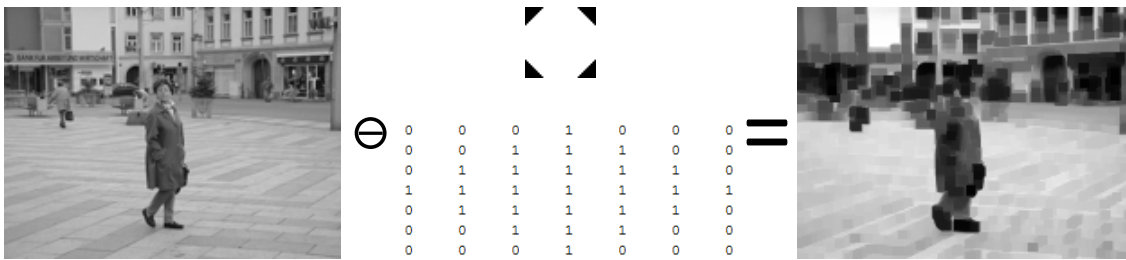


Figura 10: Ejemplo del uso de la operación morfológica de erosión usando un elemento estructural en forma de disco.

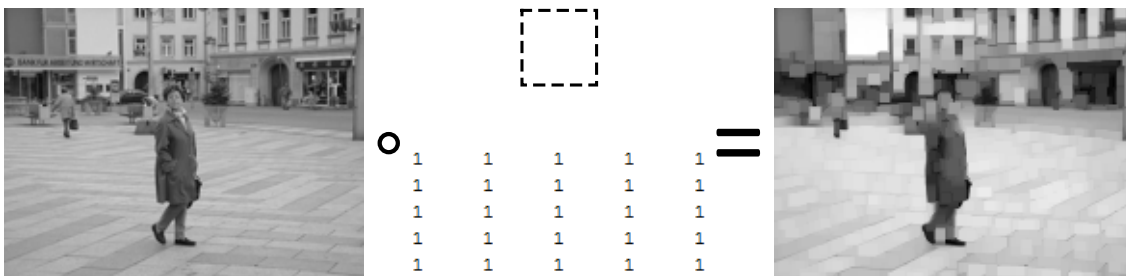


Figura 11: Ejemplo del uso de la operación morfológica de apertura usando un elemento estructural en forma de cuadro.

por la forma del elemento estructural. El efecto de esta operación, si todos los elementos del elemento estructural son positivos, es que la imagen resultante tiende a ser más oscura que la imagen de entrada y las regiones de alto brillo son más pequeñas. La Figura 10 presenta un ejemplo de esta operación usando un elemento estructural en forma de disco.

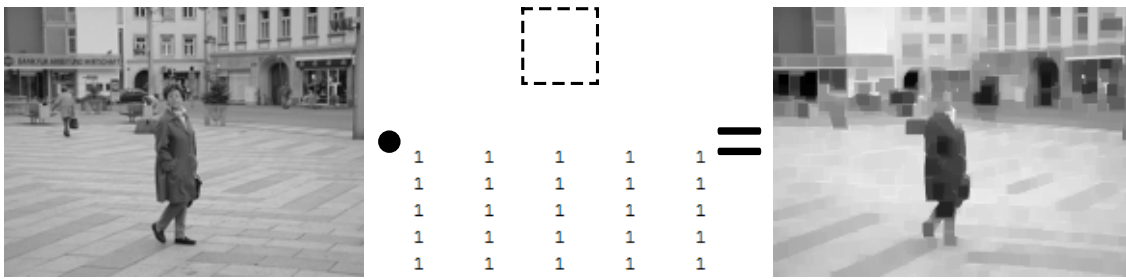
### Apertura y cerradura

Como se mencionó previamente, la erosión y dilatación son transformaciones no invertibles. Si una imagen de entrada se erosiona y luego se dilata, no se recuperaría la imagen original. Por el contrario, se obtendría una imagen con menos detalles que la imagen original. Esta combinación de erosión seguida por dilatación da pie a una transformación morfológica llamada apertura. Definida como:

$$I \circ b = (I \ominus b) \oplus b \quad (27)$$

en ambas operaciones se utiliza el mismo elemento estructural. La Figura 11 muestra un ejemplo de esta operación.

Otra operación morfológica es la cerradura. La cual resulta de aplicar una transforma-



**Figura 12: Ejemplo del uso de la operación morfológica de cerradura usando un elemento estructural en forma de cuadro.**

ción de dilatación, seguida de una de erosión, de manera formal:

$$I \bullet b = (I \oplus b) \ominus b \quad (28)$$

Al igual que en la apertura, ambas transformaciones aplican el mismo elemento estructural. En la Figura 12 se muestra el uso de esta operación.

### Añadir o eliminar

La operación añadir o eliminar forma parte de los operadores homotópicos; son aquellas transformaciones que mantiene la relación de contigüidad entre las regiones y los huecos en la imagen de entrada. Las operaciones descritas en los apartados anteriores solo han utilizado un elemento estructural  $b$ , y puntos que pertenecen a la imagen  $I$ . La operación de añadir o eliminar utiliza puntos que pertenecen al complemento de  $I$ , que no pertenecen a la imagen; además de un par de conjuntos disjuntos  $b = b_1, b_2$  conocido como elemento estructural compuesto. Esta transformación se define como:

$$I \circledast b = \{x : b_1 \subset I \wedge b_2 \subset I^c\} \quad (29)$$

esto es, para que un punto forme parte de la imagen de salida se deben cumplir dos condiciones simultáneamente. En la parte  $b_1$  del elemento estructural compuesto, cuando tiene su punto representativo sobre el punto  $x$ , debe estar contenido en  $I$ ; además, la parte  $b_2$  del elemento estructural debe estar contenido en el complemento de la imagen de entrada. Un ejemplo del resultado de esta operación se puede observar en la Figura 13.

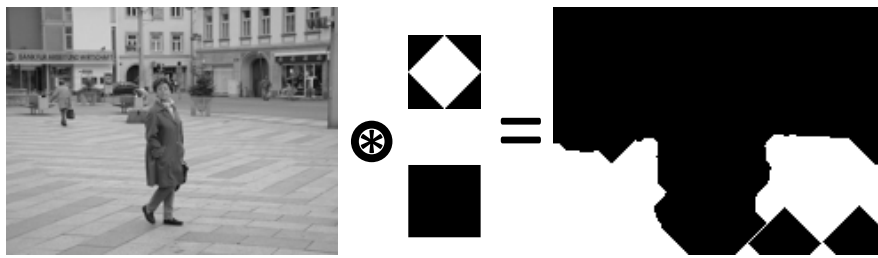


Figura 13: Ejemplo del uso de la operación morfológica de añadir o eliminar usando un elemento estructural en forma de diamante.

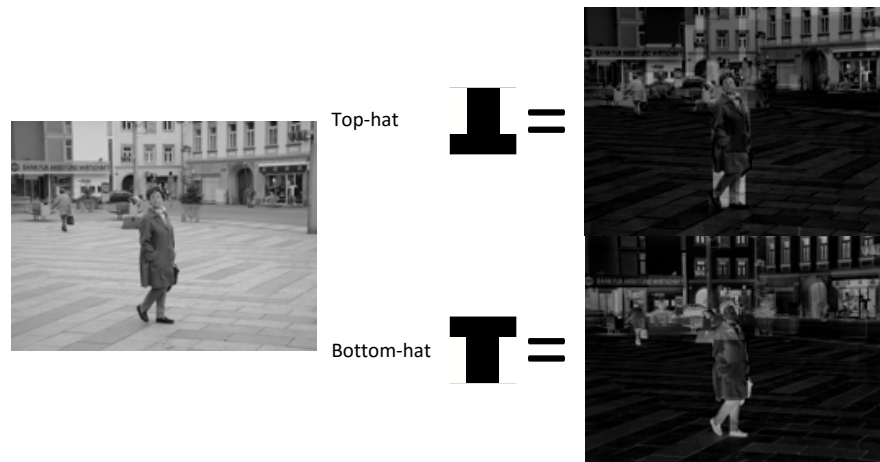


Figura 14: Ejemplo del uso de la operación morfológica de sombrero superior e inferior.

### Transformación sombrero

Otro tipo de operaciones morfológicas implementadas en este trabajo de tesis son las transformaciones de sombrero, en sus dos versiones, superior (*top-hat*) e inferior (*bottom-hat*); llamadas así por la forma del componente estructural que utilizan, el cual tiene forma de un sombrero de copa. Estas operaciones son útiles para resaltar detalles ante la presencia de sombras. Su definición formal es la siguiente:

$$I_{hat} = I - (I \circ b) \quad (30)$$

entonces, la operación de sombrero es la diferencia entre la imagen de entrada y la apertura de la imagen con el elemento estructural. La Figura 14 muestra el resultado de las operaciones de sombrero superior e inferior.



Figura 15: Ejemplo del uso de la operación morfológica de esqueleto.



Figura 16: Ejemplo del uso de la operación morfológica de perímetro.

### Esqueleto y perímetro

Los operadores sirven para extraer la estructura de los objetos. El operador de esqueleto realiza una secuencia de adelgazamientos, una operación basada en la transformación de añadir o eliminar; el objetivo del esqueleto es reducir los bordes del objeto lo más posible sin separar regiones conectadas en la imagen de entrada. Así, la transformación de adelgazamiento se define como,

$$I \triangleright b = I - (I \otimes b) \quad (31)$$

El adelgazamiento secuencial, o esqueleto, se define en base a un conjunto de elementos estructurales complejos  $b_{(1)}, b_{(2)}, b_{(3)}, \dots$  donde  $b_{(i)} = \{b_{i1}, b_{i2}\}$ . Así, el operador de esqueleto se define como,

$$I \triangleright \{b_{(i)}\} = (((I \triangleright b_{(1)}) \triangleright b_{(2)}) \dots \triangleright b_{(i)} \dots), \quad (32)$$

la Figura 15 muestra un ejemplo del uso de esta operación.

Por otra parte, la operación de perímetro aplica una secuencia de transformaciones con el objetivo de eliminar los píxeles interiores de los objetos en la imagen, y así la imagen de salida solo contendrá los contornos de los objetos (Figura 16).



En este capítulo se describieron diversos conceptos que fueron necesarios para realizar el trabajo propuesto. Se discutió, en términos generales, el paradigma de la evolución artificial. Además, se detallaron los principios básicos de la optimización multi-objetivo, bajo el cual se pretende hacer la búsqueda de los operadores sobre imágenes y evaluar su calidad en términos de varias métricas. Finalmente, se presentó el enfoque funcional que sigue el modelo propuesto, reemplazando la caracterización de los objetos por un conjunto de operaciones. Estas se pretenden construir por medio de la evolución artificial utilizando las transformaciones descritas en la última sección de este capítulo. El siguiente capítulo se enfoca en describir el modelo propuesto en esta tesis, empezando por describir los modelos que sirvieron como base para su creación.

## Capítulo 3. Corteza visual artificial multi-objetivo

---

El sistema visual humano generalmente realiza el proceso de detección como un paso necesario para resolver la tarea de clasificación. Sin embargo, no existe evidencia en cual de los dos procesos tiene mayor impacto al procesar la información visual que se percibe del ambiente. Este trabajo, se enfoca principalmente en la tarea de clasificación; no obstante, la idea es optimizar el modelo usando un enfoque multi-objetivo, al emular los dos procesos realizados por el sistema natural, como una estrategia para lograr mejores resultados de clasificación.

En esta capítulo se presentan los modelos que sirvieron como inspiración para el diseño del modelo propuesto en este trabajo de tesis. Además, se describen los modelos del estado-del-arte contra los que se compara el redimiendo de clasificación. Y finalmente, se detalla el funcionamiento de las diferentes etapas del modelo propuesto, llamada cortex visual artificial multi-objetivo (AVCMO, por sus siglas en inglés *artificial visual cortex multi-objective*).

### 3.1. Antecedentes

La vista es uno de los sentidos más importantes para los seres humanos. Ya que esta aporta aproximadamente el 70% de la información que percibe el cerebro. Esta información ayuda en la toma de decisiones, que se lleva a cabo mientras el sistema interactúa con su ambiente. Por esto, varias comunidades científicas han enfocado su investigación en entender la estructura y organización del cerebro, con el objetivo de emularlo. Existen diversos modelos computacionales Fukushima (1987); Olshausen *et al.* (1993); Walther *et al.* (2002); Itti *et al.* (1998); Riesenhuber y Poggio (1999); Serre *et al.* (2005); Mutch y Lowe (2008); Wersing y Körner (2003); Ghodrati *et al.* (2012); Huang *et al.* (2011) que se inspiran en la estructura jerárquica del sistema visual humano, sus características neurofisiológicas y diferentes teorías neuropsicológicas, tal como: la teoría de integración de características (Treisman y Gelade, 1980), la teoría de competencia sesgada (Desimone y Duncan, 1995a), el paradigma de reconocimiento por componentes (Biederman, 1987), el modelo de células simples y complejas (Hubel, 1982) y el modelo de las dos rutas corticales (Mishkin *et al.*, 1983). Estos sistemas resuelven diferentes tareas visuales tal como el reconocimiento, detección y clasificación de objetos.

A continuación se describen los principales modelos neurológicos y computacionales que sirvieron como base para el desarrollo del modelo propuesto en esta tesis.

### **3.1.1. La corteza visual**

Los ganadores del Nobel en fisiología, Hubel y Wiesel (1959), fueron los primeros en describir el flujo de información dentro de la corteza visual. En su trabajo proponen una estructura celular jerárquica. Ellos identifican las células simples que responden a estímulos visuales sencillos, tales como líneas rectas o bordes en una posición y orientación específica; y las células complejas que responden a estímulos visuales de bordes en orientaciones específicas, sin importar la posición. Además, descubren la existencia de células hiper-complejas las cuales que se excitan con la respuesta de las células complejas, las cuales corresponden a estímulos visuales como esquinas y segmentos curvos (Hubel y Wiesel, 1962; Hubel, 1982).

Más tarde, los trabajos de Ungerleider (1982) extienden la estructura jerárquica al establecer que el tratamiento de la información visual se divide en dos rutas diferentes; muestran la existencia de dos sub-sistemas y los ubican en áreas particulares dentro de la corteza visual. De esta manera, se establece el modelo de las dos rutas, la ventral y la dorsal; además, se brinda evidencia de la funcionalidad de cada uno de estos sub-sistemas. Por un lado, la ruta ventral se encarga de contestar la pregunta ¿Qué es el objeto que se observa? Mientras que la ruta dorsal se enfoca en responder la pregunta ¿Dónde se encuentra el objeto de interés?

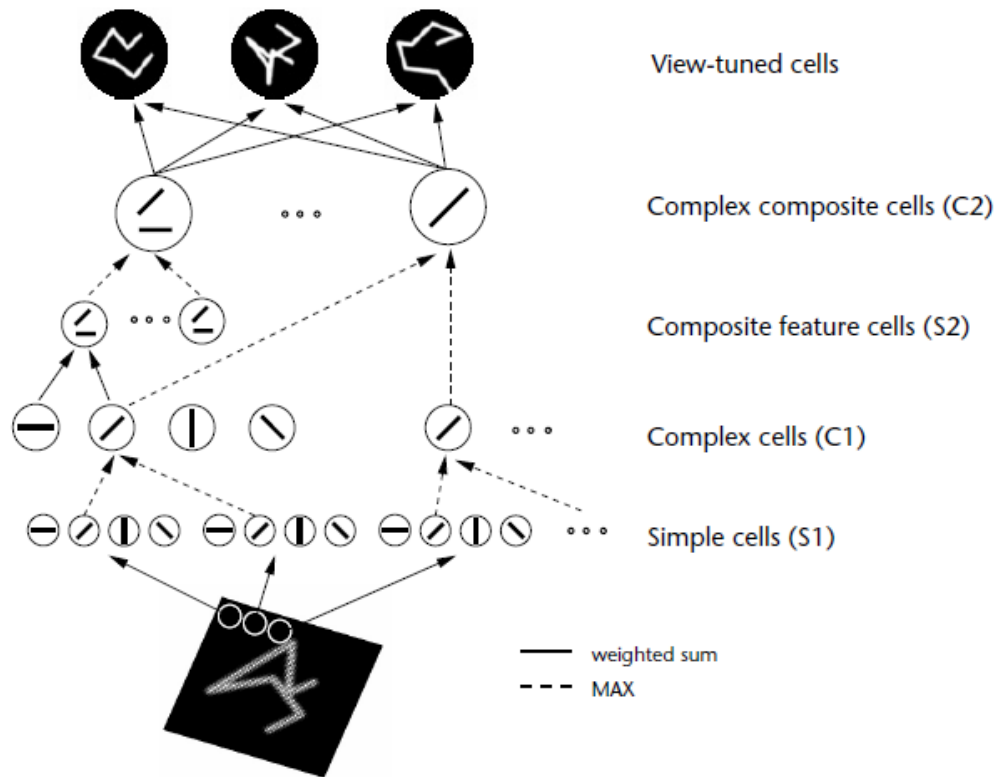
Sobre esta línea de investigación, Milner y Goodale (2006) exponen un modelo que considera a la ruta ventral como el sistema encargado de la representación abstracta del mundo visual, que además de brindar información inmediata sirve para almacenarla y usarla en percepciones futuras. Así, la ruta ventral permite crear conceptos mentales para reconocer e interpretar estímulos visuales subsecuentes. En contraste, se plantea la ruta dorsal como el sistema controlador de las acciones, estableciendo una guía para las acciones visuales al momento de ejecutarlas. Esto es, esta ruta habilita las reacciones y movimientos consecuentes de estímulos visuales, ya sea guiando la mirada hacia el objeto de interés o induciendo algún movimiento para alcanzarlo. De esta manera, se sabe que existe una distinción entre la visión para la percepción y la visión para la acción.

Cabe notar que han existido diversos modelos que integran la estructura jerárquica de ambas rutas, las operaciones y las transformaciones que sufre la información visual en ellas con el objetivo de resolver problemas de reconocimiento, clasificación y detección de objetos (Walther *et al.*, 2002; Olague *et al.*, 2014a; Clemente *et al.*, 2015). Sin embargo, el modelo expuesto en esta tesis es el primero en integrar la estructura y transformaciones, y además integrar las tareas, de detección y reconocimiento, con el objetivo de clasificar objetos. Algunos de los modelos en la literatura que han abordado las tareas de la ruta ventral y sirvieron como base para el modelo detallado en este trabajo son los estudios de Fukushima (1987); Riesenhuber y Poggio (1999); Serre *et al.* (2005); Mutch y Lowe (2008); Ghodrati *et al.* (2012); Olague *et al.* (2014a).

Desde un punto de vista computacional, el primero algoritmo, inspirado en el funcionamiento de la corteza visual, para el reconocimiento de objetos fue formulado por Fukushima (1987). Este modelo, conocido como neurocognitron, es capaz de reconocer letras y números considerando cambios de posición. El neurocognitron es una red neuronal, inspirada en el modelo jerárquico descrito por Hubel y Wiesel (1959), que consiste de una capa de entrada, seguida de una serie de estructuras modulares llamadas células S y células C; que hacen referencia a las células simples y complejas.

Sobre esta línea de investigación, Riesenhuber y Poggio (1999) proponen un modelo jerárquico con etapas similares a las enunciadas en el neurocognitron, pero agregan una operación *max* para las capas de las células complejas. Además, el modelo incluye lo que los autores llaman "diccionario de características", compuesto de un conjunto de "parches", pequeñas porciones de imágenes para la etapa de reconocimiento, en vez de una red neuronal. Este modelo se conoce como *HMAX*. La Figura 17 muestra un bosquejo del modelo; el modelo es una extensión del modelo de células complejas de Hubel y Wiesel (1962), consiste en una jerarquía de capas con operaciones lineales, siguiendo la notación de Fukushima, capas S que realizan la operación de *template matching*; y operaciones no lineales, capas C, que realizan la operación *max*.

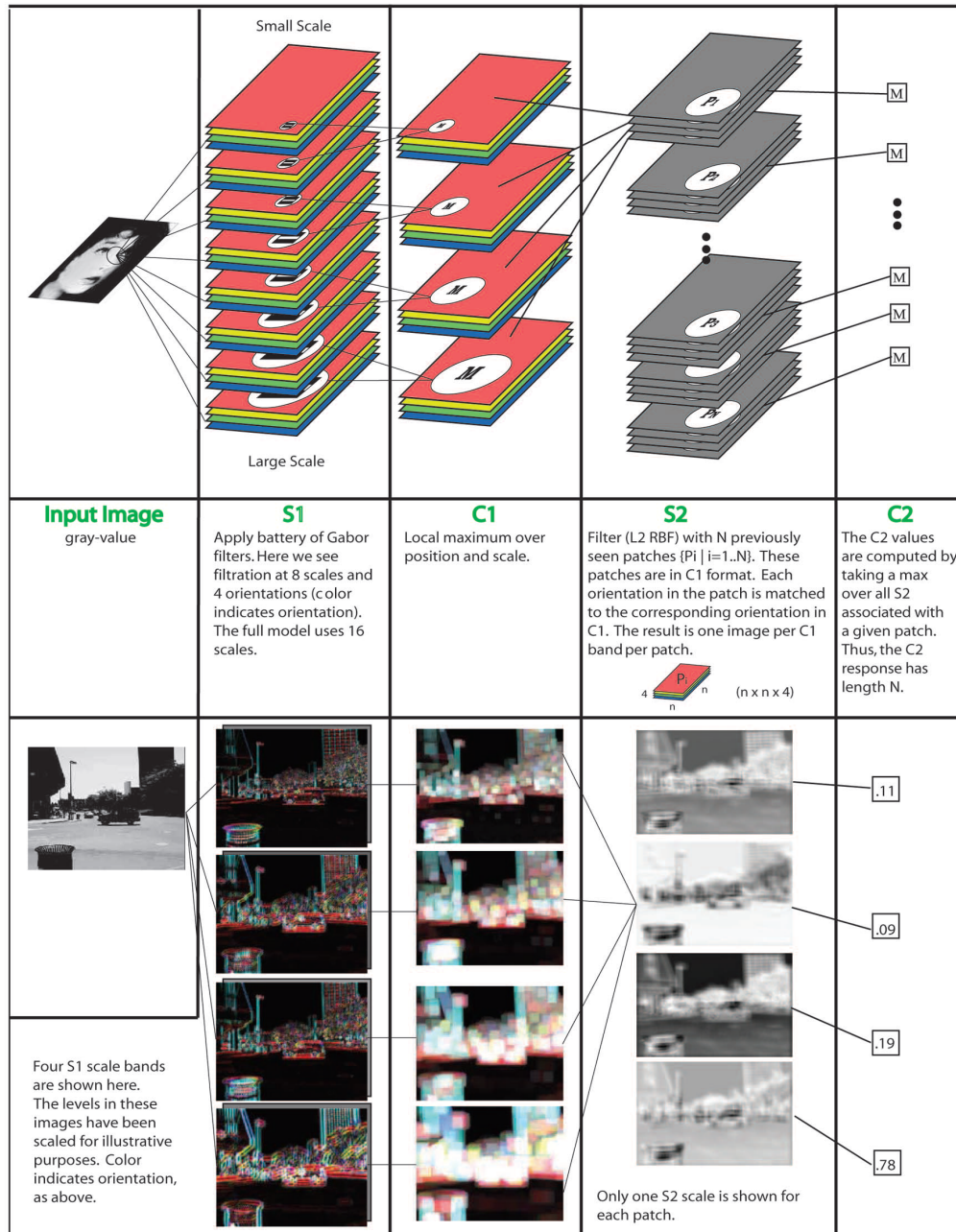
Posteriormente, Serre *et al.* (2007) proponen un sistema jerárquico que sigue la organización del cortex visual para el reconocimiento de objetos. Este modelo construye una representación de las características alternando entre operaciones de *template mat-*



**Figura 17:** Esquema del modelo de HMAX. Las capas S realizan operaciones lineales (*template matching*) y las capas C realizan operaciones no lineales (operación *max*) (Riesenhuber y Poggio, 1999).

*ching* y selección de máximos. En este trabajo se describe el concepto de "diccionario universal", un conjunto de parches, seleccionados al azar, que podría ser utilizado para reconocer una amplia gama de objetos; reconocimiento basado en forma y textura. En la Figura 18 se presenta un diagrama del modelo, se muestra como la imagen de entrada se descompone en 4 direcciones diferentes, utilizando filtros de Gabor, y a 16 escalas. Después, se realiza un muestreo sobre las imágenes resultantes a través de la operación *max*. Seguido, se comparan las imágenes con el diccionario de características y una operación *max* selecciona los estímulos más parecidos. Finalmente se utiliza una máquina de vector de soporte (SVM, por sus siglas en inglés *support vector machine*) para la clasificación.

Después, Mutch y Lowe (2008) proponen un modelo similar al de Serre *et al*; utilizando los filtros de Gabor en todas las posiciones y escalas, alternando entre las operaciones de *template matching* y máximo. Sin embargo, proponen mejoras al modelo. Se escalan las imágenes al mismo tamaño,  $140 \times 140$  píxeles; se elimina el procesamiento piramidal, ya que todas los filtros se aplican sobre la imagen de tamaño original; se presenta el concep-



**Figura 18: Diagrama a bloques del sistema propuesto por Serre *et al.* (2007). La imagen de entrada se descompone en 4 direcciones y 16 escalas, se asocian los resultados usando la operación de máximo y se realiza una comparación con cada una de las características previamente aprendidas; finalmente la clasificación se hace a través de una SVM.**

to de orientaciones dominantes, de esta manera el sistema se enfoca en las orientaciones de con mayor respuesta; además, proponen transformar las imágenes de entrada en vectores de características, similares a un vector descriptor que se clasifica usando una SVM. En la Figura 19 se muestra el flujo de información en este modelo, se muestra como la imagen de entrada se transforma en un vector de  $d$  valores, que representa la respuesta de cada una de las características previamente aprendidas. De forma similar al modelo de Serre *et al.* la transformación se lleva a cabo a través de intercalar entre operaciones de *template matching* y máximos.

En su trabajo, Ghodrati *et al.* (2012) sugieren utilizar el modelo del HMAX para el reconocimiento de objetos. La diferencia entre las propuestas radica en la construcción del diccionario de características. Mientras que en el trabajo de Riesenhuber y Poggio (1999) la selección de los parches se hace de manera aleatoria, Ghodrati *et al.* proponen reducir el número de parches en el diccionario utilizando un algoritmo evolutivo; logrando así un reconocimiento más eficiente. En la Figura 20 se muestran los diccionarios construidos por el modelo HMAX y el planteado por Ghodrati *et al.* Los diccionarios fueron creados para las imágenes de rostros, correspondientes a la base de datos de Caltech.

Finalmente, el trabajo expuesto por Clemente (2015) presenta un modelo computacional conocido como la ruta ventral artificial (AVS, por sus siglas en inglés *artificial ventral stream*). Es un modelo que se inspira de la funcionalidad de las diferentes capas en la ruta ventral y sigue la estructura jerárquica del modelo planteado por Serre *et al.* (2007). La AVS se utiliza para la clasificación multi-clase. La diferencia principal con los sistemas previos radica en el hecho que este modelo propone reemplazar el diccionario de características por un conjunto de operaciones visuales que permitan extraer la información discriminaste de los objetos de interés. La ventaja de este enfoque orientado a las funciones es la reducción de operaciones con las imágenes, esto gracias a que se elimina la necesidad de comprar todos los parches del diccionario.

Una vez descritos los diferentes modelos que se han inspirado de la ruta ventral, a continuación se discuten modelo relacionados a la ruta dorsal. Como se mencionó previamente, dicha ruta está relacionada con la tarea de localización de los objetos, la cual está estrechamente relacionada con un proceso conocido como la atención visual.

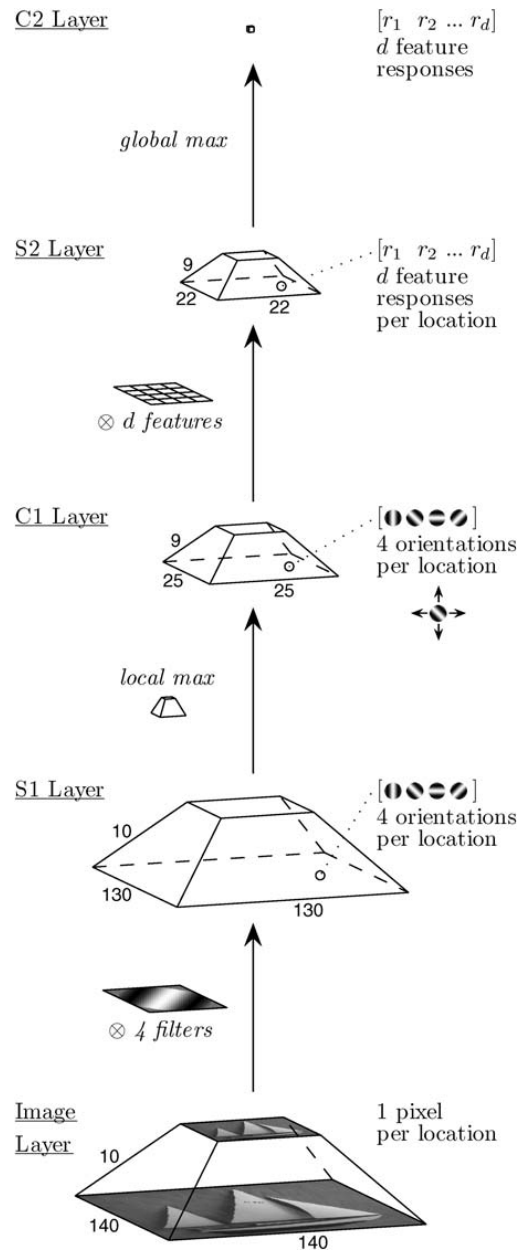


Figura 19: Diagrama del flujo de información en el modelo propuesto por Mutch y Lowe (2008).



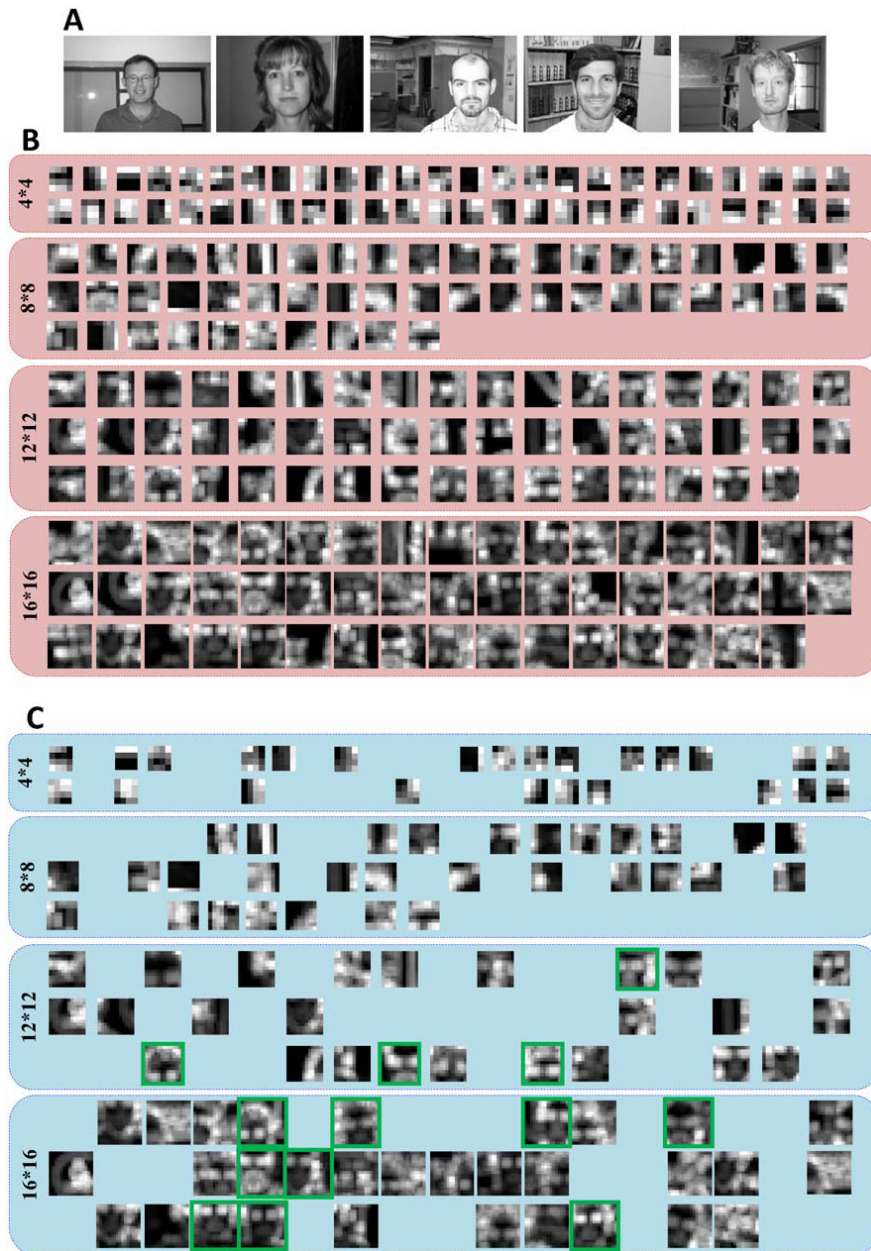


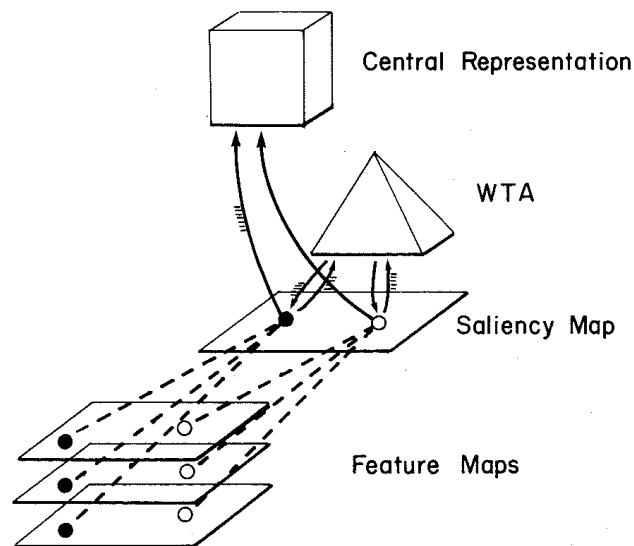
Figura 20: Comparación entre el diccionario de parches creado por el modelo HMAX y el modelo expuesto por Ghodrati *et al.* (2012).

### 3.1.2. Modelo de la ruta dorsal artificial

En esta sección se describe el modelo de la Ruta Dorsal Artificial (ADS, por sus siglas en inglés Artificial Dorsal Stream), modelo de alto interés para esta tesis dado que es uno de los pilares para la creación del modelo propuesto. La ADS es una estrategia de atención visual basada en el funcionamiento de la ruta dorsal de la corteza visual humana. El conocimiento de este procesamiento natural de información brinda la posibilidad de establecer mecanismos para la etapa de detección de objetos bajo el paradigma de la atención visual. Algunos modelos de la literatura que han abordado este problema y sirvieron como base para la creación del modelo de este trabajo de tesis son:

El trabajo de Koch y Ullman (1985) describe el primer modelo de atención. Este modelo utiliza redes de elementos similares a neuronas; además, se sugiere el análisis de las localidades del espacio visual que difieren de su entorno, en base a características elementales como color, orientación, dirección de movimiento y la disparidad. Utilizando estas características visuales, se genera una representación temprana del entorno en lo que los autores definen como mapas de características. Posteriormente, la información proporcionada por los mapas se integra a través de un proceso selectivo, creando un mapa de sobresalencia, que describe las regiones conspicuas de la escena. Finalmente, se describe una red llamada el ganador lo toma todo (WTA, por sus siglas en inglés *Winner-take-all*), mecanismo utilizado para establecer una representación central sobre la región más prominente, o la región atendida, como se muestra en la Figura 21.

Después, Itti y Koch (2001) exponen un modelo computacional para la atención visual basado en el modelo previamente descrito. Los autores describen un modelo para la atención involuntaria o abajo-arriba (*bottom-up*), es decir, el manejo de la atención basado en los estímulos que se perciben del ambiente. Este modelo supone que la sobresalencia de los estímulos depende del entorno y contexto. Además, utilizan un único mapa topográfico de sobresalencia, el cual contiene la prominencia de todos los estímulos presentes, como mecanismo de control de la atención. En su modelo, Itti y Koch, describen un proceso conocido como inhibición del retorno (IRO, por sus siglas en inglés *inhibition of return*) que se utiliza para evitar que el modelo atienda una región de la imagen previamente seleccionada. Los cambios en la regiones atendidas en una imagen están



**Figura 21: Modelo de atención visual propuesto por Koch y Ullman (1985).** Este diagrama muestra las etapas de la atención, los mapas de características como representación temprana, seguidos del mapa de sobresaliencia, que después del proceso selectivo lleva a una representación central.

relacionados al movimiento sacádico. La Figura 22 presenta un diagrama de este modelo. Se muestra la descomposición de la imagen de entrada en las diferentes dimensiones de características, las cuales son transformadas por operaciones para detectar características pre-atentas, para posteriormente integrarlas en un mapa de sobresaliencia. Dicho mapa sirve de entrada al proceso de WTA para la selección de las regiones.

Más tarde, Walther y Koch (2006) plantea un modelo computacional más sofisticado. Este modelo propone que la atención visual está relacionada al proceso de serialización de la información visual, estableciendo que es necesario enfocar un objeto para poder reconocerlo. Sin embargo, resulta complicado enfocar un objeto que no se conoce. Para esto, Walter y Koch proponen el uso de proto-objetos, unidades volátiles de información visual que pueden ser seleccionados por el proceso de atención para posteriormente validarlos como objetos. En la Figura 23 se observa la descomposición de la imagen de entrada en tres dimensiones: color, orientación e intensidades; posteriormente son procesadas por la descomposición piramidal de centro-periferia y finalmente se integran en el mapa de sobresaliencia; este mapa es la entrada al proceso de WTA que selecciona las diferentes regiones con apoyo del mecanismo de IRO.

Finalmente, el modelo de mayor interés para el desarrollo de esta tesis es el enunciado por Dozal *et al.* (2014) dado que representa una de las bases para el desarrollo

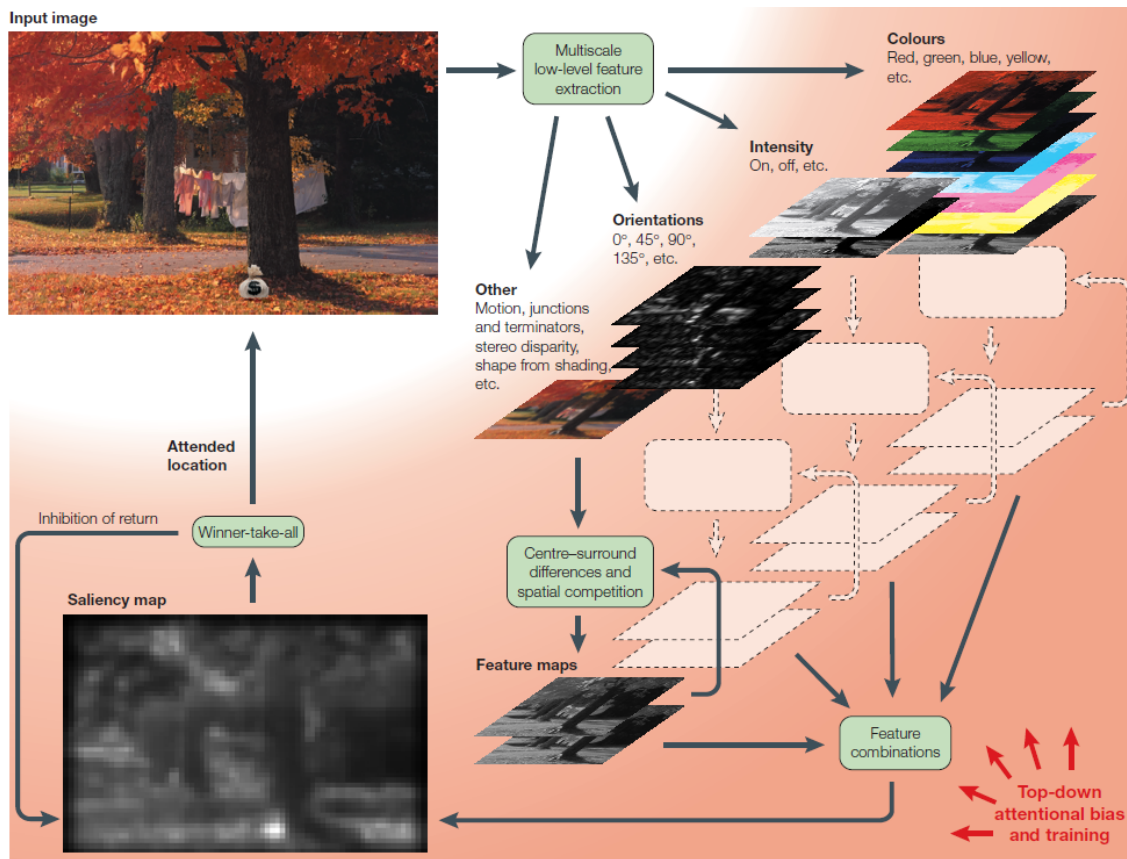


Figura 22: Modelo para el control de la atención abajó-arriba propuesto por Itti y Koch (2001).

del modelo que aquí se propone. Este modelo se conoce como la ruta dorsal artificial (ADS, por sus siglas en inglés *artificial dorsal stream*). La ADS propone un nuevo modelo jerárquico basado en el concepto de un escenario pre-atento, es decir, se considera que los elementos en la escena destacan a la vista. De esta forma, la atención visual se diseña con la intención de localizar regiones conspicuas de un objeto en particular.

La primer etapa del proceso corresponde a las transformaciones que se llevan a cabo en la corteza visual primaria (V1) cuyo proceso consta de operaciones sobre la imagen de entrada, extrayendo información de cuatro dimensiones de características: orientación, color, forma e intensidad, de acuerdo a la teoría de integración de características descrita en Dozal *et al.* (2014). Las características visuales son separables y conllevan un proceso para localizar los objetos en una escena. De esta forma a la imagen de entrada se le aplican una serie de Operadores Visuales ( $VO_s$ ) especialmente diseñados para cada dimensión con el fin de extraer información pertinente. La idea es construir estos operadores a través de un proceso de optimización. El resultado de aplicar estos  $VO_s$  sobre la imagen de entrada es un Mapa Visual ( $VM$ ) por dimensión, cada uno contiene

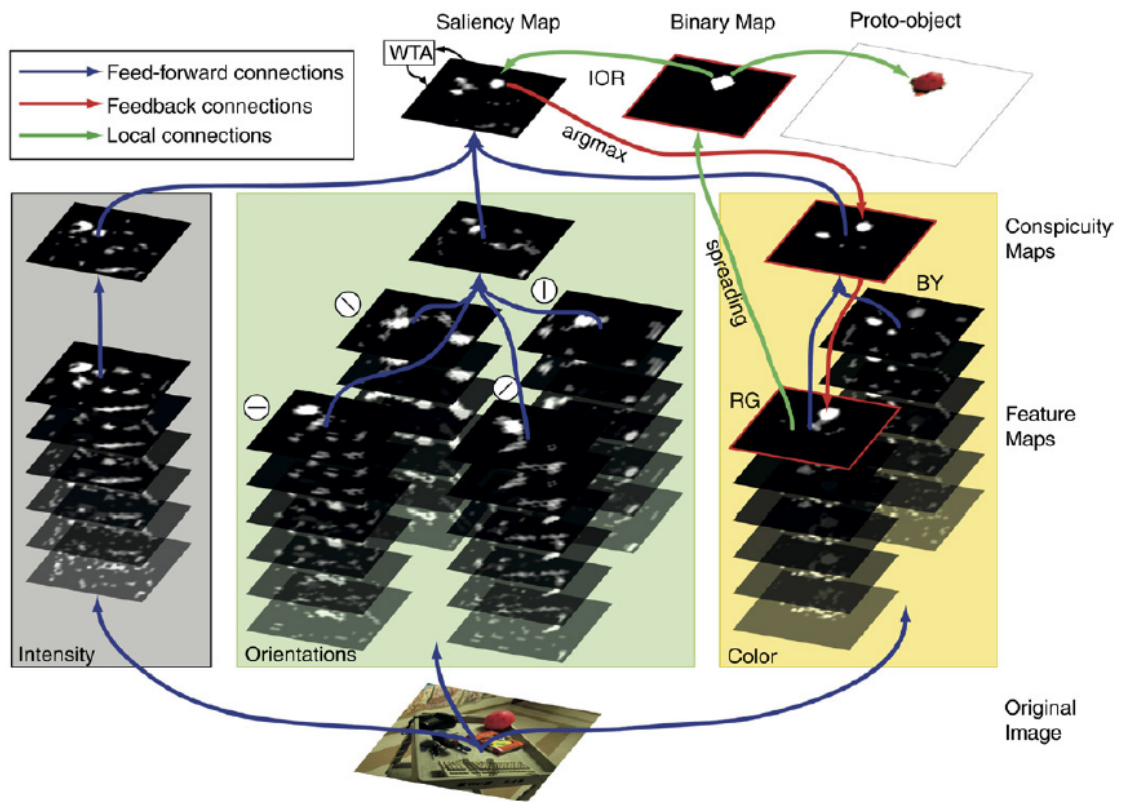
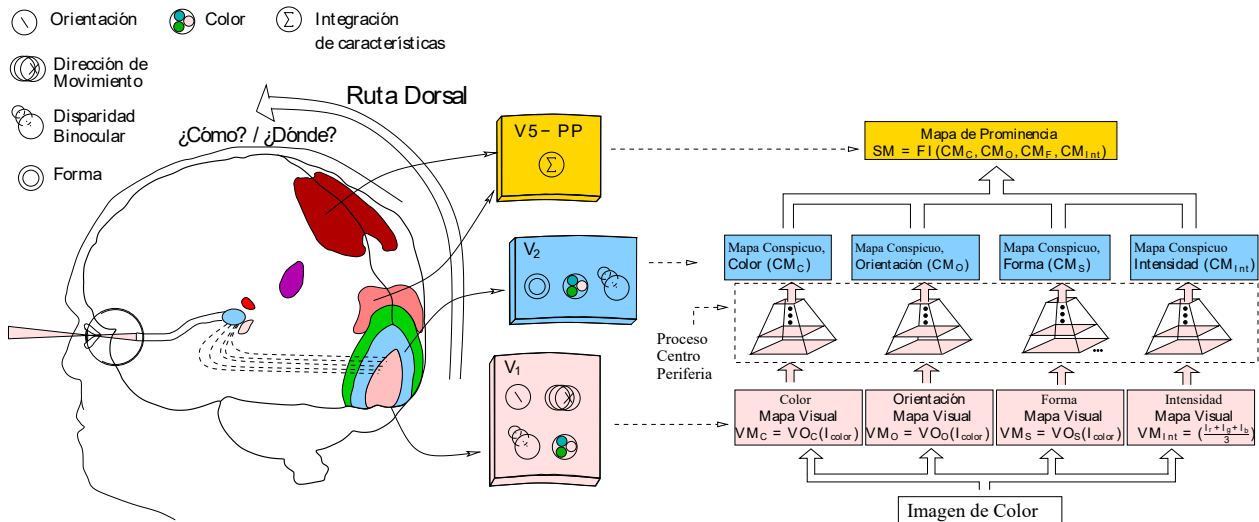


Figura 23: Ilustración de los pasos del proceso para obtener las regiones atendidas por el modelo propuesto por Walther y Koch (2006).



**Figura 24: Analogía entre sistema natural y la ADS.**

la información filtrada de la dimensión correspondiente.

El segundo proceso corresponde a la capa V2 de la corteza visual. En la cual se lleva a cabo un proceso conocido como centro y periferia, este combina de manera eficiente la información de los VMs y es útil para detectar los bordes y formas en cada una de las dimensiones. Este proceso se realiza computacionalmente aplicando un suavizado Gaussiano sobre el VM a nueve escalas, este procesamiento reduce las dimensiones del VM formando una pirámide de mapas. Posteriormente se extraen seis niveles de las pirámides, de modo que simule el procedimiento del centro y periferia del sistema biológico. El objetivo de todo lo anterior es procesar las imágenes de modo que los resultados sean independientes a cambios de escala y que todo el proceso garantice que las regiones de la imagen estén respondiendo al área indicada. Este proceso se realiza para cada una de las dimensiones de características y arroja como resultado lo que se conoce como Mapas Conspicuos ( $CM$ ).

En la tercera etapa se realiza la combinación de características, esta operación está inspirada en las transformaciones que se producen en el área cerebral correspondiente a la capa V5 de la corteza visual. De esta forma, una vez que se separa la información y se realiza el proceso para obtener los VMs y los CMs, se considera que las características se combinan de modo que se aprovechen los mapas que se calcularon con anterioridad. Este proceso de integración se realiza a través de otro VO el cual se construye también por medio de un proceso de optimización. El resultado de este proceso es un Mapa de

Prominencia (*SM*, del inglés Saliency Map) el cual contiene un proto-objeto cuyo objetivo es señalar la zona de atención de la imagen, la cual corresponde a la salida de este algoritmo. La analogía de las transformaciones que se llevan a cabo en el sistema natural asociadas a los procesos computacionales realizados por la ADS se muestra en la Figura 24.

De esta manera, el proto-objeto brindado por la salida del algoritmo de la ADS se pretende utilizar como sistema de detección del objeto de interés, y así construir un vector descriptor de dicha región.

### 3.1.3. Integración de las dos rutas

Como se mencionó previamente, la corteza visual se divide en dos sub-sistemas independientes, la ruta ventral y la ruta dorsal. Hasta ahora se han descrito modelos que se inspiran de una o de la otra para resolver las tareas de reconocimiento y detección de objetos. Al igual que el modelo descrito en esta tesis, existen modelos que han integrado la funcionalidad de ambas rutas con el objetivo de reconocer objetos.

De esta manera, un modelo computacional de interés particular para este trabajo es el conocido como la corteza visual artificial (AVC, por sus siglas en inglés Artificial visual cortex) (Olague *et al.*, 2014a), el cual se basa en el modelo HMAX presentado por Riesenhuber y Poggio (1999), ya que este modelo aborda el problema del reconocimiento de objetos inspirado en el funcionamiento del sistema visual humano. Este modelo fue propuesto por Olague *et al.* y ha mostrado buenos resultados al resolver el problema de ausencia/presencia en la tarea de reconocimiento de objetos. El AVC se basa principalmente en dos modelos: un modelo psicológico llamado teoría de integración de características (Treisman y Gelade, 1980) y un modelo neurofisiológico llamada modelos de dos rutas corticales (Mishkin *et al.*, 1983; Milner y Goodale, 2006).

La primera teoría propone que el proceso de atención visual en los seres humanos se compone de dos etapas. La primera se conoce como etapa pre-atentiva, donde la información visual se procesa en paralelo en diferentes dimensiones de características que componen la escena: color, forma, orientación, frecuencia espacial, brillo y dirección de movimiento. La segunda etapa, llamada atención focal, integra las características



extraídas en la etapa anterior, con el objetivo de subrayar una región en la imagen. Entonces, la atención visual es la capacidad de una criatura, viva o artificial, de enfocar un objeto de interés en un ambiente visual (Koch y Ullman, 1985). De esta manera, la atención visual se define como “el proceso que establece una relación entre las diferentes propiedades visuales en una escena, percibidos a través del sistema visual, con el objetivo de encontrar los características visuales necesarias para resolver la tarea que se esté realizando”(Dozal *et al.*, 2014).

La segunda teoría es el paradigma de las dos rutas corticales. Este modelo neurofisiológico establece que existen dos rutas de información dentro del cortex visual, las rutas dorsal y ventral. Ambos sub-sistemas reciben como entrada la misma información visual, sin embargo, la diferencia entre ambas rutas es el conjunto de transformaciones que sufre la información en cada una de ellas (Farivar, 2009). La ruta dorsal está relacionada principalmente a la detección de objetos en la escena y la atención visual (Desimone y Duncan, 1995b; Milner y Goodale, 2006). Por otra parte, la ruta ventral está ligada a la tarea de reconocimiento de objetos y al proceso de descripción (Mishkin *et al.*, 1983).

De esta manera, al igual que el modelo de la ADS, el modelo de la AVC se divide en etapas, la primera conocida como extracción de características y la segunda se define por el proceso de descripción de la imagen en cuestión. La Figura 25 muestra los diferentes procesos que se llevan a cabo dentro del modelo. La primera etapa es igual a la del modelo de la ADS, la imagen de entrada es procesada en cuatro dimensiones de características diferentes, color, orientación, forma e intensidad. Este procesamiento se realiza a través de cuatro operadores independientes, y al igual que en la ADS, el mapa resultante de cada operador se conoce como mapa visual.

La siguiente transformación que sufre la información visual también coincide con el modelo de la ADS. Se realiza un proceso de centro-periferia, operación que lleva a la creación de los cuatro mapas conspicuos, los cuales se distinguen por contener información prominente en cada una de las dimensiones de características. Hasta este punto se terminan las similitudes con el modelo de la ADS, y la información sufre transformaciones similares a las propuestas en los modelos que imitan la funcionalidad de la ruta ventral.

El proceso de descripción de las imágenes se inspira en el diccionario de característi-



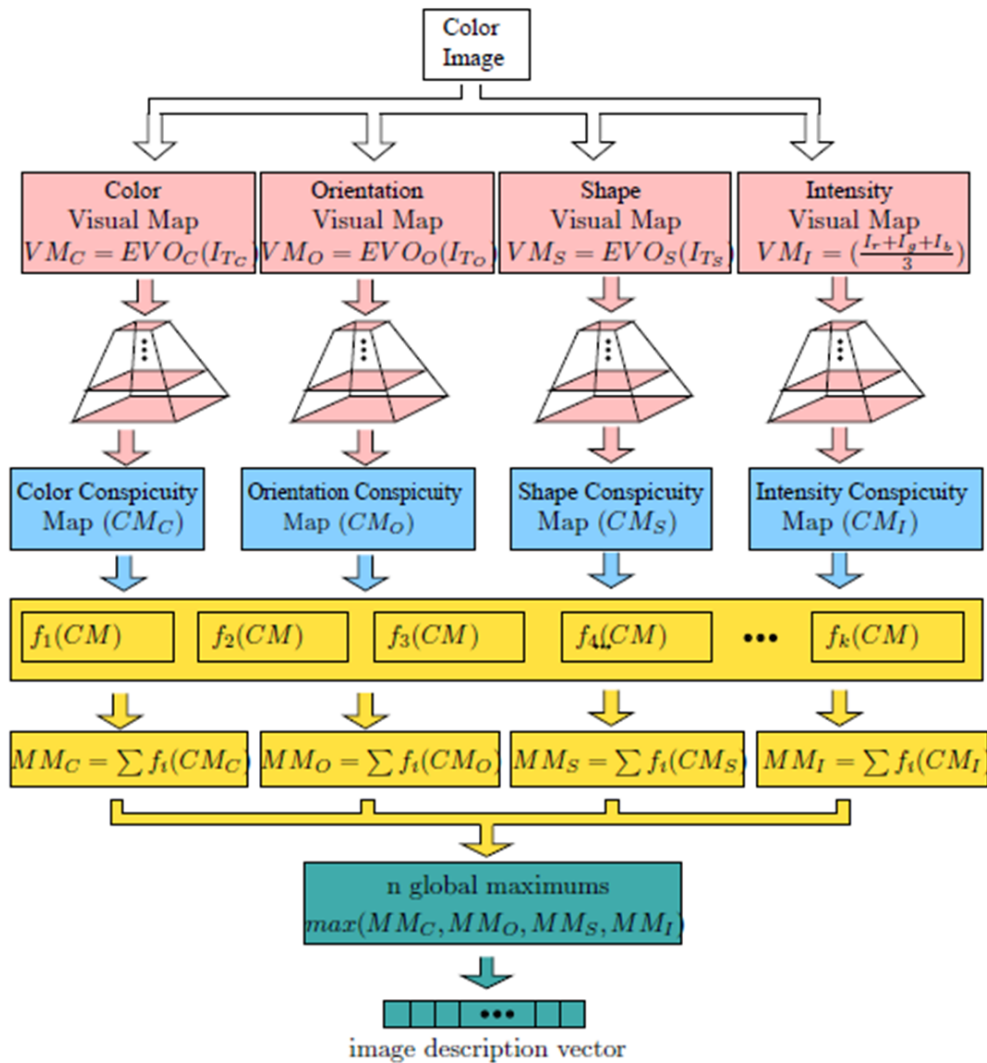


Figura 25: Modelo del AVC propuesto por Clemente (2015).

cas previamente descrito. Sin embargo, en vez de construir el diccionario a través de porciones de imágenes seleccionados al azar, como se haría en un modelo basado en los datos, el modelo de la AVC propone utilizar un conjunto de funciones que ayuden a extraer las características discriminantes en las imágenes a tratar; es decir, se propone un enfoque funcional. Así, partiendo de los mapas conspicuos, se aplica un conjunto de  $k$  operadores para la construcción de lo que se conoce como mapas mentales (MM), a diferencia de los operadores utilizados para los VM, el conjunto de operadores es el mismo para todas las dimensiones. El objetivo de estos operadores es filtrar la información visual y extraer la información que caracterice al objeto de interés. Entonces, al integrar la salida de cada uno de los operadores a través de una sumatoria, se obtiene un MM por cada dimensión.

El siguiente paso en el modelo es la construcción del vector descriptor (DV, por sus siglas en inglés *descriptor vector*) de la imagen. Esto se logra concatenando los cuatro MMs y utilizando una operación de máximos para extraer los  $n$  valores más altos, los cuales se utilizan para construir el DV. En este caso, el modelo lo utilizan para clasificar las imágenes de las base de datos de Caltech y GRAZ; logrando buenos resultados utilizando un vector descriptor de 200 elementos. Finalmente, los DV se clasifican usando una SVM.

### 3.2. Trabajo relacionado

En la actualidad existen diversos sistemas evolutivos que se enfocan en resolver los problemas previamente descritos. Por ejemplo, Olague y Trujillo describen un sistema multi-objetivo de programación genética (MOGP, por sus siglas en inglés Multi-Objective Genetic Programming) aplicado en la síntesis de operadores para la detección de puntos de interés, usados en la tarea de reconocimiento de objetos (Olague y Trujillo, 2011). Shao *et al.* proponen un sistema de aprendizaje de características para clasificar los objetos de la base de datos Caltech-101. En su trabajo, ellos evolucionan un conjuntos de operadores bidimensionales usando MOGP, para crear lo que llaman un descriptor de imagen "casi óptimo" (Shao *et al.*, 2014). Por su parte, Al-Sahaf *et al.* presenta dos métodos basados en programación genética, los sistemas "One-shot GP" "Compound-GP", que buscan evolucionar un programa para la tarea de clasificación de imágenes de texturas (Al-Sahaf *et al.*, 2015).

Otros sistemas bio-inspirados que han mostrado buenos resultados para resolver el problema de clasificación de objetos son las redes neuronales convolucionales profundas (CNN, por sus siglas en inglés Convolutional Neural Network), estos modelos se definen por un conjunto de capas jerárquicas de neuronas, donde la salida de cada neurona depende de la salida de la capa anterior, así como de un grupo de transformaciones no lineales, tal como el escalamiento. Estas redes tratan de emular las transformaciones que se llevan a cabo en el sistema visual natural, con el objetivo de generar una representación de alto nivel de la información visual. De esta manera, Schulz y Behnke presentan una arquitectura de CNN para la segmentación de imágenes de las bases de datos de GRAZ-02 y Weizmann Horses; ellos proponen agregar a la red una capa de inferencia global, la cual maneja las salidas de la red como las potencias unarias y en pares de un grafo, y así encontrar una aproximación al problema de corte s-t para lograr la tarea de segmentación (Ranftl y Pock, 2014).

Además de los sistemas previamente listados, otros sistemas de interés para este trabajo de tesis, son los trabajos presentados por Opelt *et al.* (2006); Mutch y Lowe (2008); Huang *et al.* (2011); Ghodrati *et al.* (2012). Esto se debe a que los sistemas descritos en estos trabajos se implementan para clasificar los objetos en las bases de datos de GRAZ-01 y GRAZ-02. Los modelos de Mutch y Lowe (2008); Ghodrati *et al.* (2012) ya fueron descritos en la Sección 3.1.1, a continuación se detallan los dos modelos restantes.

De esta manera, otros de los clasificadores estudiados en este trabajo, son los descritos en el trabajo de Opelt *et al.* (2006). En dicho artículo se propone un marco de trabajo genérico para el reconocimiento de objetos. Su propuesta se basa en integrar diversas metodologías de detección de regiones de interés junto con diferentes paradigmas de descripción de dichas regiones, por medio del modelo de aprendizaje llamado boosting; en específico el algoritmo de AdaBoost. El proceso de detección de regiones se propone desde dos enfoques, detección de regiones de discontinuidad, regiones alrededor de puntos de interés; y detección de regiones de homogeneidad, regiones en la imagen con valores de intensidad similares o regiones con texturas homogéneas. Para la detección de regiones discontinuas los autores utilizan detectores invariantes a escala, como el detector Harris-Laplace, y detectores de puntos de interés invariantes a transformaciones afines; además del detector basado en diferencias de Gaussianas (DoG). En cuanto a

las regiones homogéneas, se implementan diversos algoritmos de segmentación, como la segmentación basada en la medida de similitud. Por otra parte, la descripción de las regiones se realiza por medio de valores de intensidad, porciones de imagen del vecindario de los puntos de interés; momentos de intensidad, varianza, coeficiente de variación, suavidad, asimetría y curtosis; y finalmente el descriptor de SIFT.

Finalmente, el clasificador EBIM, sugerido por Huang *et al.* (2011). Este sistema es una extensión del modelo descrito por Serre *et al.* (2007). Entonces, los autores proponen agregar, al modelo descrito en 3.1.1, un mecanismo para eliminar las entradas poco informativas, imponiendo una restricción de dispersión; además de aplicar un mecanismo de retro-alimentación en la etapa de selección de características. El objetivo de la restricción de dispersión es reducir la cantidad de información que el sistema debe procesar; esto se logra al implementar un proceso de detección de puntos de interés en base al cálculo del gradiente en ambas direcciones y eliminando los píxeles con menor magnitud. Así mismo, proponen optimizar la selección de características para el proceso de clasificación; donde el sistema original selecciona los parches de manera aleatoria, este sistema proponen implementar un esquema de *boosting* para seleccionar los parches más adecuados para discriminar entre las clases.

Una vez descritos los diferentes modelos en la literatura y el estado-del-arte, a continuación se detalla el modelo base para esta tesis.

### **3.3. Metodología**

El modelo de la AVC fue diseñado para la clasificación de objetos sin importar las características de color, orientación o forma que lo identifiquen, ni las condiciones de iluminación, escala o posición en que se capturan las imágenes (Olague, 2016). Una de sus principales aportaciones es la manera en que selecciona las características visuales de la imagen con el fin de construir una representación abstracta del objeto de interés. Por consiguiente, el sistema busca puntos prominentes en la imagen para construir un descriptor de imagen, el cual se usa en el proceso de clasificación. Entonces, cuando el sistema procesa imágenes donde el objeto de interés ocupa una gran porción de la imagen, el sistema alcanza un rendimiento de clasificación de 98 % o mayor; tal es el caso en las imágenes de la base de datos de Caltech-5 (Clemente, 2015). Sin embargo, cuando

el sistema se implementa para la clasificación de imágenes donde el objeto de interés ocupa una pequeña región de la imagen y está inmerso en una escena concurrida, como es el caso de las imágenes presentadas en la bases de datos en el trabajo de Opelt *et al.* (2006), el rendimiento de clasificación de la AVC disminuye. Esto se debe a que la mayoría de las características seleccionadas para construir el descriptor de imagen se extraen de otros elementos en la escena, y no del objeto de interés. Por otra parte, el decremento en la clasificación se puede deber al hecho de que el descriptor se construye usando puntos diseminados sobre mapas de características, donde la información visual ha sido transformada por diversas operaciones; por ende, es probable que se haya perdido información del objeto de interés que podría ser útil para la construcción de un descriptor eficiente.

Cabe mencionar que a pesar que el modelo de la ADS logra buenos resultados en la tarea de detección del objeto de interés, el modelo en ningún momento se enfoca en describir el objeto encontrado; mucho menos en reconocerlo o clasificarlo. Por esta razón, dicho modelo no podría implementarse como un clasificador de la manera que fue diseñado. En esta tesis, se propone una nueva metodología para la construcción del vector descriptor de la imagen. El proceso de descripción se realiza usando una región de la imagen, en vez de usar puntos diseminados sobre la superficie de la imagen. En consecuencia, el modelo de la AVCMO es capaz de subrayar una región del objeto de interés para el proceso de descripción basado características visuales de la clase del objeto; por lo tanto, el sistema localiza el objeto dentro de la imagen de forma implícita. Además, en esta trabajo se propone un esquema de retro-alimentación para la construcción del vector descriptor, usando los mapas visuales de las etapas tempranas del sistema, en los cuales la información visual del objeto de interés solo ha sido modificada por un operador. De esta manera, las siguientes secciones se avocan a describir el funcionamiento del modelo propuesto en este trabajo.

### **3.3.1. El modelo de la corteza visual artificial multi-objetivo**

De forma similar al modelo de la AVC, el modelo de la AVCMO se divide en dos etapas principales. En la primera etapa, el sistema detecta y subraya los atributos visuales que caracterizan al objeto. Mientras que en la segunda etapa, el sistema se enfoca en encon-

trar la región más prominente de la imagen, donde el sistema supone que se encuentre el objeto de interés, y utilizarla para la creación del vector descriptor, el cual se aplica en la tarea de clasificación.

### 3.3.1.1. Detección y transformación de características visuales

La entrada para la AVCMO es una imagen digital a color representada por sus tres canales en el modelo RGB (rojo, verde, azul). La imagen es transformada posteriormente a los modelos de color CMYK (cyan, magenta, amarillo, negro) y HSV (matíz, saturación, valor), creando así el conjunto  $I_{color} = \{I_r, I_g, I_b, I_c, I_m, I_y, I_k, I_h, I_s, I_v\}$ , donde cada elemento del conjunto corresponde a un componente o canal de los modelos de color. Después, los elementos en  $I_{color}$  son transformados por cuatro operadores visuales evolutivos ( $EVO_s$ , por sus siglas en inglés *Evolved Visual Operator*) que se definen de la siguiente manera:

$$EVO_d : I_{color} \rightarrow VM_d. \quad (33)$$

Cada operador se aplica de manera independiente con el fin de destacar características visuales prominentes del objeto de interés, basado en cuatro dimensiones de características específicas: color ( $C$ ), orientación ( $O$ ), forma ( $S$ ) e intensidad ( $Int$ ), estos operadores se muestran en la Figura 26. De esta forma,  $d$  es un elemento del conjunto de dimensiones  $d \in \{C, S, O, Int\}$ . Así, cada mapa visual ( $VM_d$ ) contiene información prominente del objeto de interés sobre las diferentes dimensiones de características.

Después del cómputo de los mapas visuales, todos estos son sujetos a un proceso de centro-periferia, cuyo objetivo es emular el funcionamiento de la zona receptiva de las células ganglionares en la retina, esta zona permite a las células transmitir información no solo de las células fotorreceptivas que captan la luz, sino información de las células a su alrededor; lo que permite generar información de contraste (Walther *et al.*, 2002). Así, el objetivo de este proceso es generar un mapa conspicuo ( $CM_d$ ) por cada dimensión, y funciona de la siguiente manera. Esta sub-rutina se describe por dos pasos; primero, empezando por los mapas visuales  $VM_d$ , se crea un pirámide Gaussiana de ocho niveles de la forma:

$$P_d^\sigma = \{P_d^{\sigma=0}, P_d^{\sigma=1}, P_d^{\sigma=2}, \dots, P_d^{\sigma=8}\} \quad (34)$$

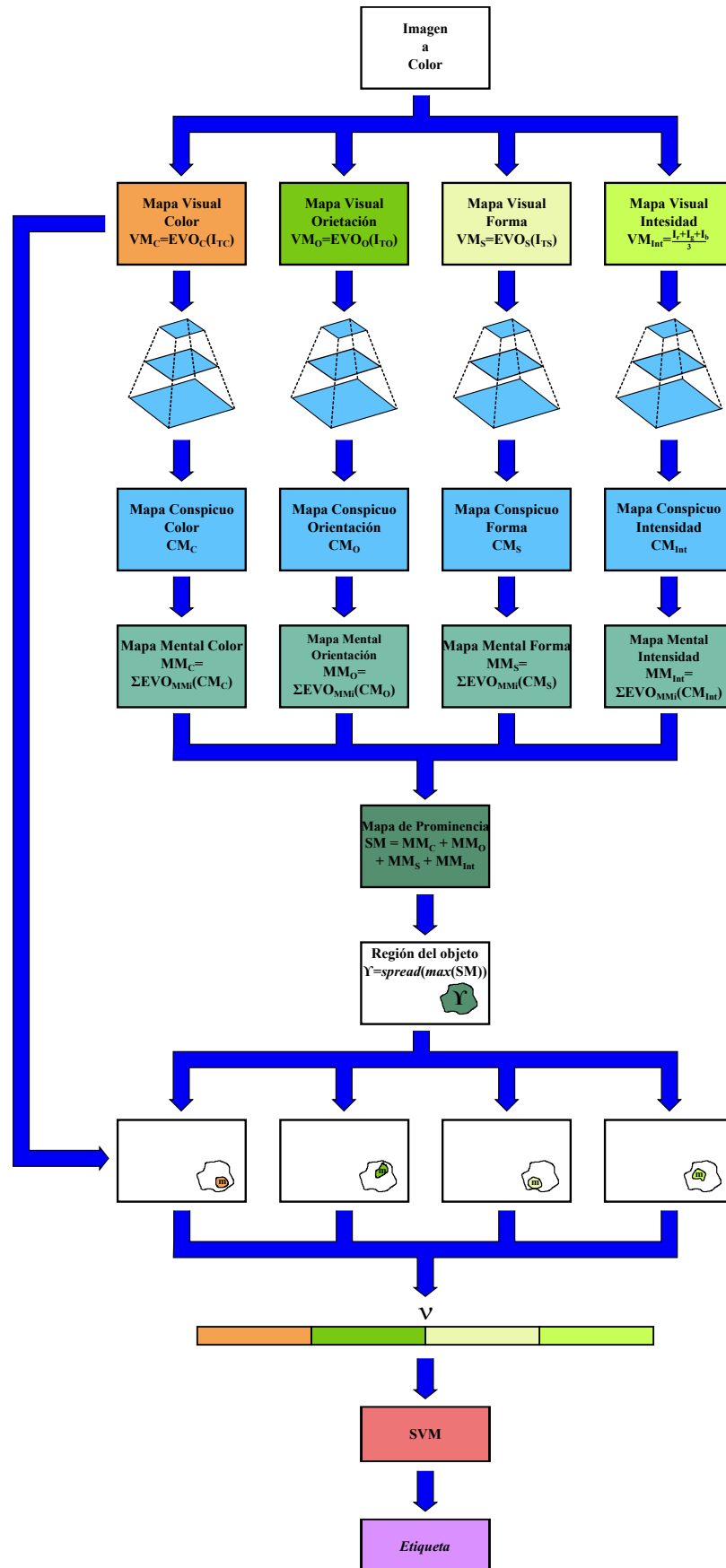


Figura 26: Diagrama de flujo de la información visual en el modelo propuesto AVCMO.

donde  $\sigma$  denota el difuminado Gaussiano que define cada nivel en la pirámide, y representa la reducción de escala de la imagen original. La segunda etapa en este proceso utiliza esta pirámide para generar seis mapas como sigue:

$$Q_d^j = P_d^{\sigma = \lfloor \frac{j+9}{2} \rfloor + 1} - P_d^{\sigma = \lfloor \frac{j+2}{2} \rfloor + 1}, \quad (35)$$

donde  $j = 1, 2, \dots, 6$ . Dado que los niveles de  $P_d^\sigma$  tienen tamaños diferentes, todos los mapas son redimensionados, por medio de una interpolación por el vecino más cercano, al mismo tamaño que el mapa más pequeño. Después, estos seis mapas son normalizados y se suman para crear un mapa único. El mapa resultante se normaliza nuevamente y se redimensiona al tamaño del mapa entrada  $VM_d$ ; así, el nuevo mapa define lo que se conoce como mapa conspicuo  $CM_d$ , un mapa para cada una de las dimensiones de características.

### 3.3.1.2. Detección y descripción del objeto

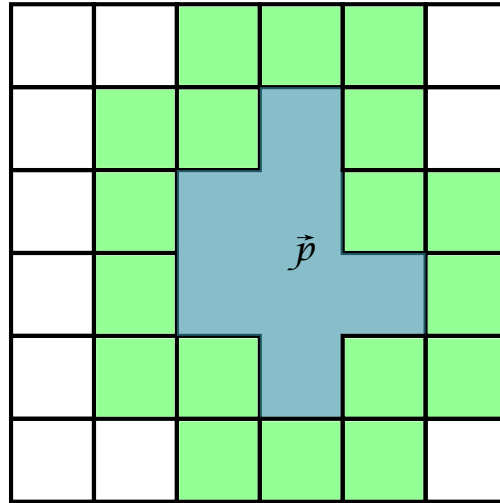
La segunda etapa del modelo de la AVCMO pretende encontrar la región de la imagen más prominente usando los mapas conspicuos. Idealmente, esta región debe contener información visual relevante acerca del objeto de interés, con el objetivo de crear un vector descriptor de la imagen. En esta etapa del sistema se aplica un conjunto de operadores visuales con el objetivo de crear un mapa mental ( $MM_d$ ) por dimensión, como se observa en la Figura 26. Así, un conjunto de operadores  $EVO_{MM}$  se aplican sobre cada mapa  $CM_d$ , para filtrar las características más prominentes del objeto de intereses en cada uno de las dimensiones. Estos operadores se escriben de la siguiente manera:

$$MM_d = \sum_{i=1}^k (EVO_{MM_i}(CM_d)); \quad (36)$$

donde  $d$  es la dimensión de características y  $k$  representa la cardinalidad del conjunto  $EVO_{MM}$ . Entonces, la salida de todos los operadores en  $EVO_{MM}$  se integran a través de una sumatoria, creando el mapa  $MM_d$ .

Una vez que los mapas mentales son calculados, el sistema los normaliza entre 0 y 1





**Figura 27: Representación gráfica del proceso iterativo para agregar el valor máximo en el vecindario de los elementos de  $\vec{p}$ . La región azul son los puntos que forman  $\vec{p}$ , mientras que la región verde es el vecindario donde se busca el siguiente elemento.**

usando interpolación bi-lineal,

$$MM_d = \frac{MM_d - \min(MM_d)}{\max(MM_d) - \min(MM_d)}. \quad (37)$$

Después, se integran en un mapa único conocido como mapa de prominencia ( $SM$ , *salience map*) de la siguiente manera,

$$SM = MM_C + MM_O + MM_F + MM_{Int} \quad (38)$$

Partiendo del  $SM$ , las coordenadas del valor máximo en el mapa se guardan en un vector de posición  $\vec{p}$ . Enseguida, se realiza un proceso de propagación en torno de la posición almacenada, esta operación consiste en agregar, de forma iterativa, el elemento mayor en el vecindario 8-conectado alrededor de los elementos de  $\vec{p}$ , el cual se agrega como nuevo elemento  $\vec{p}(i)$ , como se observa en la Figura 27. De esta manera, se agrega  $n$  elementos al vector  $\vec{p}$  que definen una región  $\mathcal{I}$  sobre el mapa de prominencia, esta región establece el área de la imagen donde se debería encontrar el objeto de interés.

A pesar que  $\mathcal{I}$  define la posición del objeto, los valores que se usarán para construir el vector descriptor se extraen de mapas creados en la primera etapa del modelo. Para esto, la región  $\mathcal{I}$  se proyecta sobre los mapas visuales con el fin de extraer las mejores características visuales para cada dimensión. Entonces, se selecciona el pixel con mayor

valor dentro de la región proyectada en el  $VM_d$ . Una vez más se realiza una operación de propagación en busca de los  $m$  elementos mayores en cada uno de los mapas visuales, donde  $m = n/4$ . Finalmente, los  $m$  valores de cada dimensión se concatena, creado así el llamado vector descriptor  $\vec{v}$  de tamaño  $n$ . Este se toma como la entrada para una máquina de vector de soporte (SVM) para realizar la clasificación. El proceso de construcción del vector descriptor se detalla en el Algoritmo 1 y se muestra en la Figura 26.

**Algoritmo 1:** Este algoritmo describe los pasos necesarios para la construcción del vector descriptor partiendo del  $SM$  y los  $VM_d$ .

```

Entrada :  $SM, VM_d$  /*Mapa de prominencia y los cuatro mapas visuales */
Salida :  $\vec{p}, \vec{v}$  /*Región atendida de la imagen y el vector descriptor */
1  $\vec{p}[1] \leftarrow \text{coordMaxVal}(SM)$ 
2  $SM[\vec{p}[1].x, \vec{p}[1].y] \leftarrow 0$ 
3 para  $i \leftarrow 2$  a  $n$  incrementa 1 hacer
4    $\vec{p}[i] \leftarrow \text{coordMaxValNeighbor}(SM, \vec{p})$ 
5    $SM[\vec{p}[i].x, \vec{p}[i].y] \leftarrow 0$ 
6 fin
7 para cada  $VM_d$  hacer
8    $\vec{p}_d[1] \leftarrow \text{coordMaxValReg}(VM_d, \vec{p})$ 
9    $v_d[1] \leftarrow \text{getVal}(VM_d, \vec{p}_d[1])$ 
10   $VM_d[\vec{p}_d[1].x, \vec{p}_d[1].y] \leftarrow 0$ 
11  para  $i \leftarrow 2$  a  $n/4$  incrementa 1 hacer
12     $\vec{p}_d[i] \leftarrow \text{coordMaxValNeighbor}(VM_d, \vec{p}_d)$ 
13     $v_d[i] \leftarrow \text{getVal}(VM_d, \vec{p}_d[i])$ 
14     $VM_d[\vec{p}_d[i].x, \vec{p}_d[i].y] \leftarrow 0$ 
15     $\vec{v} \leftarrow \text{concat}(\vec{v}, v_d)$ 
16  fin
17 fin
18 devolver  $\vec{p}, \vec{v}$ 

```

Es importante notar que la función *coordMaxVal* regresa las coordenadas del pixel con el valor mayor en todo el mapa, mientras que *coordMaxValReg* busca el elemento máximo dentro de la región definida por el vector  $\vec{p}$ . Además, *coordMaxValNeighbor* es un algoritmo voraz que funciona sobre un vecindario de Moore de los elementos de  $\vec{p}$ .

Una vez descritos las diferentes transformaciones que componen el modelo propuesto, el siguiente capítulo se dedica a la descripción del paradigma evolutivo llamado programación cerebral, utilizado para optimizar las operaciones dentro del AVCMO, y así poder clasificar los objetos deseados. El algoritmo utilizado en este trabajo extiende el descrito en el trabajo de Dozal *et al.* (2014), agregando el enfoque multi-objetivo.

## Capítulo 4. Programación de cerebros artificiales con enfoque multi-objetivo

---

Ahora que se conoce a detalle el funcionamiento del modelo de la AVCMO, es evidente que uno de los aspectos más complicados para crear un modelo eficiente en la clasificación, es encontrar el conjunto de operadores que mejor interactúen entre ellos, así como con el resto de las transformaciones del modelo, para crear un descriptor discriminador con respecto a objetos de otras clases. De esta manera, la idea es utilizar la evolución artificial para encontrar un conjunto óptimo de operadores que consideren el modelo completo, en vez de optimizar individualmente los operadores para cada una de las etapas del sistema. Así, cada operador es una pequeña porción de un sistema complejo, inmerso en la estructura jerárquica.

### 4.1. Programación de cerebros multi-objetivo

La programación de cerebros (BP, por sus siglas en inglés *brain programming*), es un paradigma evolutivo cuyo objetivo es optimizar modelos complejos, ajustando los operadores dentro del mismo. Esta metodología es útil para modelos como el de la AVCMO, donde varios operadores independientes se integran dentro de una estructura jerárquica pre-definida. BP sigue el ciclo evolutivo de la programación genética, pero utiliza una representación multi-árbol heterogénea para los individuos, además de implementar operaciones evolutivas de cruce y mutación correspondientes a esta representación. Es importante notar que los operadores *per se* no son una solución. De hecho, un solo operador no es necesariamente interesante, dado que cada operador tiene sentido al funcionar junto con el resto de los operadores que representan una solución dentro de la estructura jerárquica. El propósito de cada operador se hace evidente al momento de interactuar con el resto, así como con las otras transformaciones del modelo AVCMO.

#### 4.1.1. Trabajos previos: Enfoques mono-objetivo

Los primeros intentos de evolucionar un modelo de la corteza visual artificial se presentaron en los trabajos de Clemente *et al.* (2012); Dozal *et al.* (2012). En ese trabajo se presenta el paradigma de BP con un enfoque mono-objetivo. El objetivo de este sistema

es evolucionar modelos de la corteza visual artificial para clasificar las imágenes de la base de datos Caltech-101. Este modelo logra buenos resultados usando la clasificación como función objetivo. Los modelos del trabajo de Olague *et al.* construyen el vector descriptor usando puntos dispersos sobre toda la imagen; debido a las condiciones de las imágenes que constituyen dicha base de datos, la mayoría de los puntos seleccionados para el vector descriptor pertenecen al objeto de interés, ya que este ocupa casi toda la superficie de la imagen. Esta condición no es cierta en las bases de datos GRAZ-01 y GRAZ-02. Por ende, es necesario integrar un nuevo criterio para la evaluación de los individuos, la cual proporcione una región de interés para crear el vector descriptor, en vez de utilizar toda la imagen.

Los autores, Dozal *et al.* (2014), presentan una implementación del algoritmo de BP para resolver el problema del foco de atención (FOA, por sus siglas en inglés *Focus of Attention*). Los autores proponen optimizar modelos de la ruta dorsal artificial (ADS, por sus siglas en inglés *Artificial Dorsal Stream*) para detectar un objeto en particular. En este trabajo, la medida-F se utiliza como función objetivo para contar el número de píxeles que se detectaron adecuadamente. Su trabajo presenta excelentes resultados para detectar el objeto blanco, pero se enfoca únicamente en localizar el objeto. El sistema resalta la región más prominente en la imagen, la cual se establece como la posición del objetivo. Sin embargo, el sistema no considera un proceso de descripción o caracterización del objeto para poderlo clasificar o reconocer.

Cabe notar que ambos enfoques mono-objetivo logran buenos resultados al resolver las tareas correspondientes. Sin embargo, los dos métodos presentan deficiencias al resolver tareas más complejas o en ambientes más complejos. Primero, la eficiencia del modelo AVC se reduce al intentar clasificar objetos que ocupan una región pequeña de la imagen. Segundo, el modelo del FOA es capaz de encontrar objetos inmersos en un ambiente concurrido, pero no extrae suficiente información del objeto para lograr clasificarlo. Entonces, una mejora plausible sería combinar los dos objetivos, detección y descripción, en un solo modelo. A través de un algoritmo de optimización que considere ambas tareas, para sintetizar modelos capaces de abordar el problema de clasificación en imágenes más complejas; como las imágenes presentes en la base de datos de GRAZ.

## 4.2. Evaluación multi-objetivo

La función objetivo mide la calidad de las soluciones al resolver el problema correspondiente. Esta tesis se enfoca principalmente en el problema de clasificación, utilizando la tarea de localización como un mecanismo que permita explorar otras regiones del espacio de soluciones, con el objetivo de encontrar mejores modelos. De esta forma, partiendo de las características del modelo, se propone el uso de dos funciones objetivo, una para medir la calidad de los modelos en la tarea de clasificación y otra que determine la calidad de las soluciones al momento de localizar al objeto de interés en la imagen.

La medida que se seleccionó para medir el rendimiento en clasificación se conoce como tasa de error igual (EER, por sus siglas en inglés *Equal Error Rate*). Esta métrica establece la probabilidad de que un algoritmo decida si dos instancia corresponden a una misma clase (Reid, 2004). El EER se define como el valor que satisface  $fpr = fnr$ ; donde  $fnr$  es la razón de falsos negativos (*false negative rate*) y  $fpr$  representa la razón de falsos positivos (*false positive rate*), y se cumple la siguiente restricción:  $fnr = 1 - tpr$ , dado que  $tpr$  es la razón de verdaderos positivos (*true positive rate*). Usando una curva de característica operativa del receptor (ROC, por sus siglas en inglés *Receiver Operating Characteristic*), el EER se puede calcular extendiendo una línea desde el punto (0,1) al (1,0), el punto donde esta recta se cruza con la curva de clasificación representa el EER. Entonces, dado un individuo  $\Phi$ , un conjunto de imágenes  $I_n$  y una función definida como:  $\Gamma : AVCMO(\Phi, I_n) \rightarrow EER$ , la función objetivo se puede escribir de la siguiente manera:

$$Objetivo_1 = \arg \min_{AVCMO(\Phi, I_n)} \frac{1}{\Gamma}. \quad (39)$$

De esta manera, el primer objetivo consisten en encontrar un individuo que maximice el 1-EER. Este objetivo mide la calidad de clasificación de cada solución al procesar un conjunto de imágenes dadas.

El segundo objetivo se basa en calcular la correspondencia entre la posición real del objeto en la imagen y la región  $\mathcal{Y}$  estimada por el modelo. En este caso, se utiliza la medida-F definida por:  $F_\alpha(\rho, \vartheta) = \frac{(1 + \alpha) \cdot (\rho \cdot \vartheta)}{(\alpha \cdot \rho) + \vartheta}$ , donde  $\alpha$  establece el balance entre precisión  $\rho$  y sensibilidad  $\vartheta$ , así  $0 \leq \alpha \leq \infty$ . Si  $\alpha < 1$  entonces  $\rho$  es mayor que  $\vartheta$ ; en

caso contrario, si  $\alpha > 1$  entonces  $\vartheta$  es mayor. Finalmente, cuando  $\alpha = 1$ , se dice que la precisión y la sensibilidad están balanceadas.

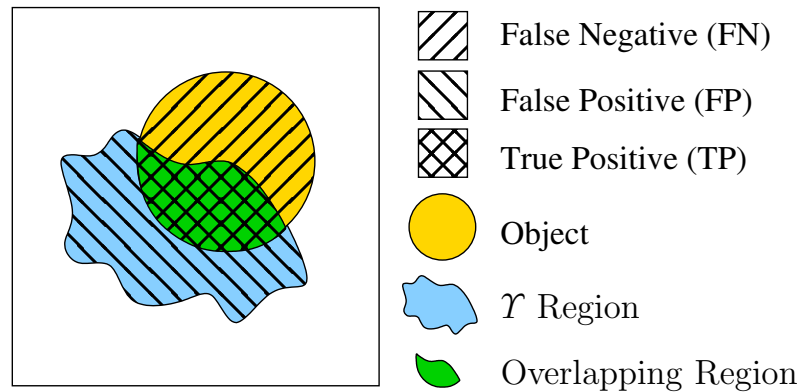
En este trabajo se considera que  $\alpha = 1$ . Los elementos considerados verdaderos positivos corresponden a los pixeles que pertenecen a la región  $\mathcal{Y}$  definida por el modelo y a la región real del objeto, mientras que los falsos positivos son los puntos en la región  $\mathcal{Y}$  que no corresponden al objeto de interés, y finalmente los falsos negativos son los puntos en el objeto que no están incluidos en la región  $\mathcal{Y}$ , una descripción gráfica de estos elementos se puede observar en la Figura 28.

De esta manera, después de procesar  $\omega$  imágenes que contienen el objeto, la calidad de la solución para la tarea de localización se define por el promedio de la medida-F sobre las  $\omega$  imágenes, y se expresa de la siguiente manera:  $\bar{F} = \frac{1}{\omega} \sum_{i=1}^{\omega} \left( \frac{2 \cdot (\rho \cdot \vartheta)}{\rho + \vartheta} \right)$ . Por lo tanto, la idea es encontrar un individuo que maximice  $\bar{F}$ . Dado un arreglo de  $EVO_s$   $\Phi$ , un conjunto de imágenes  $I_\omega$  y  $\Lambda$ , una función que se escribe como  $\Lambda : AVCMO(\Phi, I_\omega) \rightarrow \bar{F}$ ; desde una perspectiva de minimización, la función objetivo se escribe como:

$$Objective_2 = \arg \min_{AVCMO(\Phi, I_\omega)} \frac{1}{\Lambda}. \quad (40)$$

Así, el segundo objetivo consiste en encontrar una solución que maximice la medida-F promedio  $\bar{F}$  correspondiente a los pixeles detectados como parte del objeto de interés en un conjunto de imágenes al minimizar la ecuación 40. Este objetivo busca medir la calidad del modelo para detectar el objeto de interés.

Debido al enfoque multi-objetivo adaptado en este trabajo para el proceso evolutivo, operación de selección no es tan simple como en el caso de la optimización mono-objetivo, esto se debe a la diferencia en la definición de la mejor solución. En los procesos de un solo objetivo, el concepto de la mejor solución se define trivialmente en un espacio unidimensional; mientras que en un ambiente multi-objetivo, la definición del mejor individuo se basa en una relación de dominancia entre las soluciones, al evaluarlas en un espacio multi-dimensional (Olague y Trujillo, 2012). De esta manera, se han propuesto diversos sistemas para encontrar un conjunto de soluciones óptimas en el sentido de Pareto (Lin y Lin, 2014b; Behnamian, 2014). En este trabajo optamos por utilizar el siste-



$$\rho = \frac{TP}{TP+FP},$$

$$\vartheta = \frac{TP}{TP+FN}$$

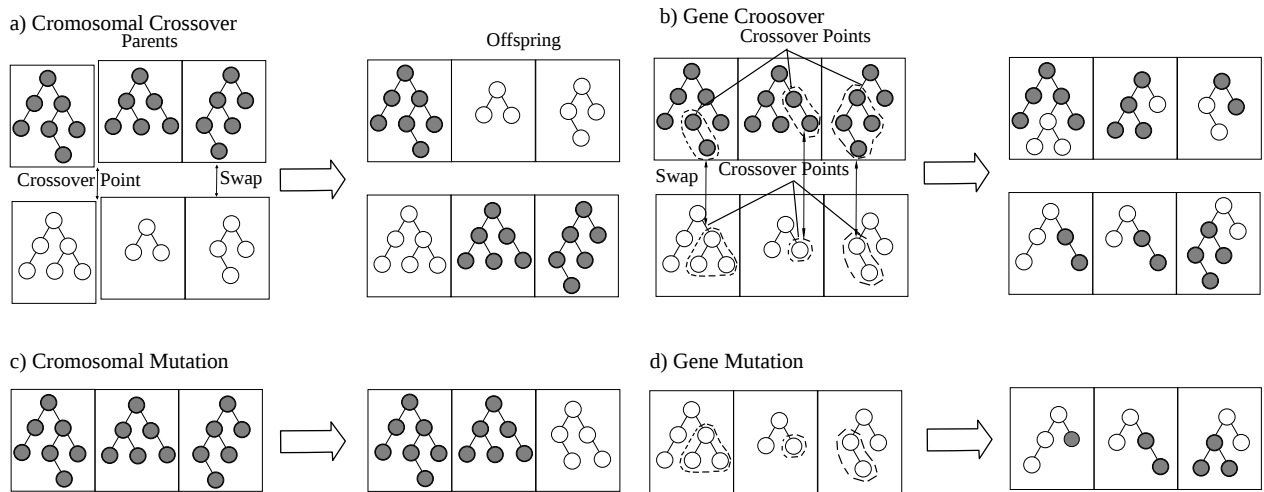
**Figura 28:** Comparación entre la región de la imagen atendida por el modelo, y la región ocupada por el objeto de interés. Utilizadas para evaluar la precisión  $\rho$  y la sensibilidad  $\vartheta$ .

ma conocido como algoritmo evolutivo de fuerza del Pareto 2 (SPEA2, por sus siglas en inglés *Strength Pareto Evolutionary Algorithm 2*)(Zitzler *et al.*, 2001) como el sistema de selección.

El algoritmo SPEA2 considera una relación de dominancia entre los individuos. Así, a cada individuo  $i$  se le asigna un valor de fuerza  $S(i)$ , que representa el número de soluciones que este domina. Entonces, la aptitud  $R(i)$  de un individuo se determina usando la fuerza de aquellos individuos de la población que lo dominan. El SPEA2 utiliza el algoritmo de agrupamiento de los  $k$ th vecinos más cercanos como mecanismo para preservar la diversidad, esto se logra penalizando a los individuos que se agrupan en una región del espacio del objetivo. Además, el algoritmo implementa un enfoque de almacenamiento de tamaño fijo y un esquema de trunque para promover la diversidad, al eliminar individuos que se encuentran muy cerca de sus vecinos. Finalmente, el sistema implementa elitismo con el fin de preservar las soluciones que forma el frente no dominado, el cual resulta ser una buena aproximación al frente Pareto (Olague y Trujillo, 2012).

### 4.3. Genotipo

Una característica importante de los *EVOs* es su independencia, esto facilita su representación computacional como un arreglo. Así, esta representación sigue la siguiente analogía con un sistema biológico. El arreglo de operadores es similar a un cromosoma. Entonces, cada operador puede ser considerado como un gen, donde cada función y ter-



**Figura 29:** Los operadores genéticos se implementan en dos niveles; la Figura (a) muestra la operación de cruce a nivel cromosoma, y la (b) a nivel gen; mientras que la (c) y (d) representan la operación de mutación a nivel cromosoma y nivel gen respectivamente.

minal que lo componen son análogos a los nucleótidos que constituyen el gen. Entonces, esta manera de representar las soluciones tiene tres niveles, el primero considera todo el arreglo, el segundo nivel consiste en observar cada operador de manera independiente; y el tercer nivel está definido por las funciones que constituyen los operadores, la Figura 29 muestra estos niveles de manera gráfica. En este trabajo, los individuos consisten de hasta 15 operadores, tres de estos se utilizan en la primera etapa del modelo, para la construcción de los mapas visuales, y hasta 12 operadores que se aplican para la creación de los mapas mentales. De esta manera, el fenotipo de dichas soluciones consiste en implementar este conjunto de operadores dentro de la estructura jerárquica de la corteza visual artificial.

#### 4.4. Operadores genéticos

Esta representación de los individuos requiere de operadores de recombinación no convencionales, debido a los diferentes niveles de organización. Por esta razón, se implementaron las operaciones de cruce y mutación descritas a continuación.

##### 4.4.1. Cruce

En el proceso evolutivo utilizado en este trabajo existen dos operaciones de cruce, una para el cruce a nivel cromosoma y otra para cruce a nivel gen:

- *Cruce a nivel cromosoma*: el objetivo de este operador es lograr la combinación



genética e intercambio de información entre dos individuos; este se logra al intercambiar segmentos del arreglo de operadores que constituyen cada uno de los individuos seleccionados para el cruce, a estos individuos se les conoce como padres. El método utilizado en esta tarea se conoce como cortar y empalmar (en inglés, *cut and splice*). Primero, se selecciona un punto de cruce de manera aleatoria del padre más pequeño, el arreglo con menos operadores, y se selecciona el mismo punto de cruce en el otro padre. Segundo, se crea un nuevo individuo, conocido como *hijo1*; partiendo del punto de cruce, se selecciona el lado izquierdo del *padre1* y el lado derecho del *padre2*. De manera similar, el *hijo2* se construye usando el lado izquierdo del arreglo del *padre2* y el lado derecho del *padre1*. Este proceso se puede observar en la Figura 29a.

- *Cruce a nivel gen*: este proceso se enfoca en las funciones que componen los genes. Se selecciona un punto de cruce en cada uno de los árboles en los individuos padres. Cabe notar que si el número de operadores en los padres difiere, el padre con el arreglo más grande pasará directamente algunos operadores al hijo que genere. Entonces, el *padre1* se selecciona para crear el *hijo1*; así, en cada uno de los operadores del *padre1* se reemplaza los sub-árboles debajo del punto de cruce con los sub-árboles debajo de los puntos de cruce en los operadores del *padre2*. De manera similar, el *hijo2* se construye reemplazando los sub-árboles de los operadores por los sub-árboles del *padre1*. De esta manera se crean dos nuevos individuos. Este proceso se puede observar en la Figura 29b.

#### 4.4.2. Operadores de mutación

Una vez que los nuevos individuos son creados, estos podrían ser modificados por uno de dos tipos de operadores de mutación: mutación a nivel cromosoma y mutación a nivel gen:

- *Mutación a nivel cromosoma*: este operador consiste en reemplazar cada operador que constituye el cromosoma con un operador creado de manera aleatoria. Esta operación se muestra gráficamente en 29c

- *Mutación a nivel gen*: en esta operación de variación se selecciona un punto de variación para cada árbol semántico; después, el sub-árbol debajo del punto de mutación se reemplaza con un sub-árbol aleatorio. Por lo tanto, este tipo de mutación solo cambia una porción de cada operador. Esta operación de mutación está representado en la Figura 29d.

#### 4.4.3. Funciones y terminales

En el modelo AVCMO cada  $EVO$  es independiente, y se construyen usando conjuntos diferentes de funciones y terminales, como se muestra en la Tabla 2. Por esta razón, se seleccionó un conjunto de funciones para cada dimensión de características, con el objetivo de encontrar las características visuales que identifiquen al objeto de interés.

Siguiendo este esquema, para el operador de orientación  $EVO_O$  las funciones seleccionadas fueron filtros de suavizado Gaussiano con valores de  $\sigma = 1$  y  $\sigma = 2$ , además de derivadas derivadas de primer y segundo orden en las direcciones de  $x$  y  $y$ .

Mientras tanto, el operador de color  $EVO_C$  se puede construir usando las funciones listadas en la Tabla 2, destacando las funciones siguientes: diferencias de color ( $Op_{r-g}(I)$ ,  $Op_{b-y}(I)$ ) y la función complemento ( $(A)^c$ ).

De manera similar, el modelo implementa la morfología matemática con el objetivo de encontrar información prominente en base a características de forma del objeto, algunas funciones son: dilatación ( $A \oplus SE_x$ ), erosión ( $A \ominus SE_x$ ), apertura ( $A \odot SE_s$ ), cerradura ( $A \odot SE_s$ ), además de transformaciones que resultan de la combinación de estas cuatro funciones básicas; este conjunto de funciones se utilizan para la construcción del operador  $EVO_S$ .

El conjunto de terminales que se utilizan para crear los operadores son los bandas de color de los diferentes modelos, que definen el conjunto  $I_{color}$ , además de la salida de algunas funciones aplicadas sobre los elementos de este mismo conjunto.

La selección de estos conjuntos de funciones se hizo de forma empírica, son resultado de diversos trabajos realizados en el laboratorio de EvoVisión (Trujillo y Olague,

**Tabla 2: Funciones y terminales para la construcción de los operadores visuales evolutivos  $EVOs$ .**

Funciones para $EVO_O$	Terminales para $EVO_O$
$A + B, A - B, A \times B, A/B,  A ,  A + B ,  A - B , \log(A), (A)^2, \sqrt{A}, k + A, k - A, k \times A, A/k, \text{round}(A), \lfloor A \rfloor, \lceil A \rceil, \text{inf}(A, B), \text{sup}(A, B), G_{\sigma=1}(A), G_{\sigma=2}(A), D_x(A), D_y(A), \text{thr}(A)$	$I_r, I_g, I_b, I_c, I_m, I_y, I_k, I_h, I_s, I_v, D_x(I_x), D_{xx}(I_x), D_y(I_x), D_{yy}(I_x), D_{xy}(I_x)$
Funciones para $EVO_C$	Terminales para $EVO_C$
$A + B, A - B, A \times B, A/B, \log(A), \exp(A), (A)^2, \sqrt{A}, (A)^c, \text{thr}(A)$	$I_r, I_g, I_b, I_c, I_m, I_y, I_k, I_h, I_s, I_v, \text{Op}_{r-g}(I), \text{Op}_{b-y}(I)$
Funciones para $EVO_S$	Terminales para $EVO_F$
$A + B, A - B, A \times B, A/B, k + A, k - A, k \times A, A/k, \text{round}(A), \lfloor A \rfloor, \lceil A \rceil, A \oplus SE_d, A \oplus SE_s, A \oplus SE_{dm}, A \ominus SE_d, A \ominus SE_s, A \ominus SE_{dm}, Sk(A), \text{Perim}(A), A \otimes SE_d, A \otimes SE_s, A \otimes SE_{dm}, T_{\text{hat}}(A), B_{\text{hat}}(A), A \circ SE_s, A \bullet SE_s, \text{thr}(A)$	$I_r, I_g, I_b, I_c, I_m, I_y, I_k, I_h, I_s, I_v$
Funciones para $EVO_{MM}$	Terminales para $EVO_{MM}$
$A + B, A - B, A \times B, A/B,  A + B ,  A - B , \log(A), (A)^2, \sqrt{A}, G_{\sigma=1}(A), G_{\sigma=2}(A), D_x(A), D_y(A)$	$MC_d, D_x(MC_d), D_{xx}(MC_d), D_y(MC_d), D_{yy}(MC_d), D_{xy}(MC_d)$

2008; Perez y Olague, 2009; Dozal *et al.*, 2014; Clemente *et al.*, 2015). La construcción de un individuo del modelo AVCMO usando la representación propuesta requiere de la optimización de varios operadores, por esto el espacio de soluciones para el modelo resulta de la combinación del espacio factible de cada uno de los operadores. Así, el espacio de búsqueda para el algoritmo se define como el número de posibles soluciones que se pueden lograr al combinar todos los posibles operadores visuales. Entonces, el tamaño del espacio de búsqueda se puede obtener analizando el conjunto de funciones y terminales descritos en la Tabla 2. En consecuencia, dada una estructura de árbol  $i$ ,  $nT$  terminales,  $nF_U$  funciones unarias y  $nF_B$  funciones binarias; el número de posibles operadores visuales  $nVO_i$  se puede calcular a través de la relación:  $nVO_i = nT^{nl_n} \times nF_U^{n_{pn_1}} \times nF_B^{n_{pn_2}}$ , donde,  $nl_n$  es el número de nodos hoja,  $n_{pn_1}$  es el número de nodos padres con al menos un hijo, y  $n_{pn_2}$  es el número de nodos padre con al menos dos nodos hijos. Por lo tanto, el tamaño del espacio de búsqueda  $S_s$  se puede calcular multiplicando la cantidad de posibles combinaciones de operadores visuales para tres dimensiones de características y los mapas mentales. Esto se puede escribir como:  $S_s = \sum_{i=1}^l (nVO_C)_i \times \sum_{i=1}^l (nVO_S)_i \times \sum_{i=1}^l (nVO_O)_i \times \left( \sum_{i=1}^l (nVO_{MM})_i \right)^k$ , donde  $l$  está dado por la profundidad de los operadores visuales y  $k$  es el número de mapas visuales. Por ende, si

se consideran las funciones y terminales de la Tabla 2,  $l = 15$ , y  $k = 12$ , el tamaño del espacio de búsqueda es alrededor de  $4.72 \times 10^{87}$  soluciones. Esto es parte de la complejidad para encontrar una solución eficiente para la clasificación de los objetos deseados.

## Capítulo 5. Experimentos y resultados

---

Una vez descrito el funcionamiento del modelo, así como el sistema evolutivo que se utiliza para optimizarlo, es necesario probar el rendimiento del modelo en un prueba real; y además esta prueba es un estándar internacional y permite comparar el rendimiento del modelo propuesto con modelos del estado del arte.

En este trabajo se aborda el problema de clasificación desde una perspectiva de ausencia/presencia, o clasificación binaria. Con este fin seguimos un protocolo de pruebas que consta de tres pasos; los primeros dos pasos definen la etapa de entrenamiento del modelo, mientras el que tercer paso corresponde a la etapa de prueba. Por esta razón se necesita tres conjuntos de imágenes para los experimentos, una por etapa. Este protocolo se detalla a continuación:

1. *Entrenamiento*: esta etapa del protocolo consiste en evaluar cada una de las soluciones con el conjunto de imágenes etiquetado como entrenamiento; se aplica el modelo para crear un descriptor de imagen para cada elemento del conjunto de entrenamiento. Después, estos descriptores se utilizan para entrenar un modelo de una máquina de vector de soporte (SVM, por sus siglas en inglés *support vector machine*); este clasificador etiqueta cada uno de los descriptores, estableciendo así cuales pertenecen a la clase y cuales no.
2. *Validación*: el objetivo de esta etapa es evitar el sobre entrenamiento. En este paso, se evalúa el rendimiento de cada una de los individuos usando otro conjunto de imágenes, llamada validación. Así, los descriptores de las imágenes de este conjunto son clasificados usando la SVM creada en el paso anterior. El sistema evolutivo utiliza los resultados, de clasificación y detección, que logran los individuos en estas imágenes como medida de aptitud.
3. *Prueba*: una vez terminado el proceso de optimización usando BP, los individuos de la última generación junto con su modelo de SVM se evalúan utilizando el tercer conjunto de imágenes, llamado conjunto de prueba. Esta evaluación define el rendimiento de los modelos, y este valor es utilizada para compararlos con otras técnicas.

## 5.1. Base de datos de imágenes

La base de datos utilizada en este trabajo es el set de imágenes GRAZ, compuesto por dos grupos conocidos como GRAZ-01 y GRAZ-02, fue propuesta por Opelt *et al.* (2006). El conjunto GRAZ-01 está constituido por dos clases: la clase bicicleta, de la cual se proporcionan 373 imágenes que contienen bicicletas; y la clase persona, para la cual se tienen 460. Además de estas imágenes, la base de datos contiene 210 imágenes que contienen bicicletas y personas, y 273 de fondo; es decir, imágenes que no contienen ni bicicletas ni personas, estas imágenes se utilizan para definir el conjunto no-clase. Una de las críticas principales a este conjunto de imágenes, es que las fotos de cada clase fueron tomadas en entornos similares; lo cual puede resultar en que el sistema clasificador utilice información del ambiente para discriminar entre las clases. Entonces, GRAZ-02 fue creada con la idea de incrementar la complejidad de las imágenes, y así brindar una prueba de clasificación con mayor diversidad. Esto se logró al integrar una tercera clase a la base de datos, además de un mejor balance entre los entornos en que fueron capturadas las fotos para cada una de las clases. Así, GRAZ-02 se compone de tres clases, donde 311 imágenes pertenecen a la clase persona, 365 fotos son de la clase bicicleta, 420 representan la clase carro y 380 imágenes corresponden al conjunto de fondo o no-clase. Este último grupo no contiene imágenes de personas, bicicletas o carros.

En este trabajo se siguió el mismo protocolo de pruebas presentado en el trabajo de Opelt *et al.* (2006), con el objetivo de comparar los resultados de clasificación del modelo propuesto con otros del estado del arte, usando el mismo número de imágenes de entrenamiento y de prueba. Por lo tanto, se seleccionaron 150 imágenes como el conjunto de entrenamiento, 75 imágenes fueron seleccionadas para validación y 75 imágenes para la etapa de prueba; estos conjuntos fueron seleccionados para cada una de las clases de manera independiente. Además de presentar un esquema estándar de comparación para la tarea de clasificación, esta base de datos contiene imágenes segmentadas para cada una de las clases, las cuales fueron utilizadas como marco de referencia para la evaluación de la medida-F.

**Tabla 3: Parámetros utilizados en el proceso evolutivo.**

Parámetros	Descripción
Generaciones	30
Tamaño de la población	400 individuos
Inicialización	Incrementando mitad-y-mitad
Cruce a nivel de cromosoma	0.4
Cruce a nivel de gen	0.4
Mutación a nivel de cromosoma	0.1
Mutación a nivel de gen	0.1
Profundidad del árbol	Selección de profundidad dinámica
Profundidad máxima dinámica	7 niveles
Profundidad real máxima	9 niveles
Selección	SPEA2
Elitismo	Frente no dominado
Tamaño del descriptor	400

## 5.2. Validación cruzada y comparación con otros métodos

Como se mencionó previamente, se implementó un sistema evolutivo de BP para optimizar las soluciones para cada una de las clases. Los parámetros evolutivos utilizados para dicho proceso se presentan en la Tabla 3.

Estos parámetros fueron definidos de manera empírica. Además de los parámetros evolutivos, es necesario establecer el número de puntos que se utilizan para caracterizar cada imagen, para estos experimentos se optó por usar un vector descriptor de tamaño  $n = 400$ .

También es necesario definir los parámetros para el sistema de selección SPEA2, para los experimentos de esta tesis son los siguientes: tamaño de la población  $\alpha = 400$ , el número de padres seleccionados para reproducción es  $\mu = 200$ , y se optó por crear  $\lambda = 200$  individuos hijos, usando un tamaño de torneo de 2 para el proceso de selección de pareja.

El sistema de BP está implementado en MATLAB, y se basa en el funcionamiento de la paquetería GPLAB presentada en la investigación de Silva y Almeida (2003). Todos los experimentos fueron realizados usando una estación de trabajo Dell Precision T7500 con procesador Intel Xeon de ocho núcleos, con 8 GB de memoria RAM y usando la tarjeta de video nvidia Quadro 4000; todo corriendo sobre el sistema operativo OpenSUSE Linux.

En esta tesis nos enfocamos en la clasificación de las cinco clases presentes en la base de datos de GRAZ debido a que presentan un escenario de clasificación complicado, además que es considerado por la comunidad científica como una prueba estándar para la tarea de clasificación. Cabe notar que el costo computacional de cada corrida evolutiva es elevado, cada ejecución dura aproximadamente 24 horas corriendo en paralelo usando los ocho procesadores de la estación de trabajo.

De esta manera, se realizó un proceso de validación cruzada con el objetivo de comprobar la eficacia del sistema propuesto. Por esta razón, se crearon 4 conjuntos de 75 imágenes para cada una de las clases en las dos bases de datos, y así poder realizar un proceso de validación de 4 dobleces (*4-fold*). Así, se seleccionaron dos grupos de imágenes para el proceso de entrenamiento, uno para la validación y uno para la etapa de prueba, como se mencionó previamente; aunado a esto, en cada iteración del proceso se intercambiaron los conjuntos de entrenamiento con el objetivo de tener cuatro ejecuciones diferentes del proceso evolutivo usando diferentes conjuntos de entrenamiento: 1 y 2, 2 y 3, 3 y 4, 1 y 4; y usando los dos sets restantes para validación y prueba. Esta metodología comúnmente se utiliza para hacer un estudio estadístico de la eficiencia de un proceso de optimización. Además, en este trabajo se ejecutó 8 veces el ciclo de los cuatro dobleces, resultando en 32 ejecuciones del proceso evolutivo para cada una de las clases. Los resultados de dichas ejecuciones se presentan en las Tablas 4, 5, 6, 7 y 8.

De esta manera, al final de cada ejecución del proceso evolutivo, los individuos de la última generación se ponen a prueba al clasificar las imágenes correspondientes al conjunto de pruebas. El individuo de la última generación que logre el mejor rendimiento en la clasificación del conjunto de pruebas, se considera como el mejor resultado encontrado en cada uno de los 32 ejecuciones, las evaluaciones correspondientes a dichos individuos son los números listados en las Tablas 4, 5, 6, 7, 8; además, los números resaltados corresponden a la solución que logra el mejor rendimiento de clasificación para cada una de las clases. Para las clases de GRAZ-01, las mejores soluciones se encontraron en la 8va corrida del 2do doblez para la clase Bicicleta y en la 2da corrida del 2do doblez. En cuanto a las clases de GRAZ-02, la solución con el mejor rendimiento se encontró en la 5ta corrida del 4to doblez para Bicicleta, 7ma corrida del 1er doblez en Persona; y finalmente, en la 1ra corrida del 3er y 4to doblez para la clase Carros.



**Tabla 4: Rendimiento de los mejores individuos para todas las ejecuciones para la clase Bicicleta de GRAZ-01.**

GRAZ-01 Bike class												
	Corr 1			Corr 2			Corr 3			Corr 4		
Doblez	Ent.	Val	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	79.00	70.00	72.00	82.00	72.00	78.00	79.00	78.00	82.00	73.67	76.00	72.00
2	79.00	70.00	80.00	77.00	76.00	76.00	80.00	78.00	82.00	72.00	82.00	72.00
3	79.00	68.00	66.00	78.00	73.00	74.00	67.00	78.00	76.00	76.00	74.00	76.00
4	80.00	68.00	76.00	74.00	70.00	76.00	73.00	78.00	80.00	71.00	86.00	72.00
Promedio	79.25	69.00	73.50	77.75	72.75	76.00	74.75	78.00	80.00	73.17	79.50	73.00
$\sigma$	0.500	1.155	5.972	3.304	2.500	1.633	6.021	0.000	2.828	2.186	5.508	2.000

	Corr 5			Corr 6			Corr 7			Corr 8		
Doblez	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	75.00	76.00	70.00	66.00	64.00	68.00	71.00	82.00	70.00	74.00	76.00	76.00
2	74.00	62.00	72.00	76.00	72.00	70.00	77.00	86.00	78.00	<b>81.00</b>	<b>72.00</b>	<b>84.00</b>
3	79.00	78.00	68.00	78.00	82.00	68.00	74.00	80.00	74.00	74.22	62.00	80.00
4	83.00	78.00	70.00	76.00	78.00	72.00	71.00	76.00	72.00	64.00	60.00	74.00
Promedio	77.75	73.50	70.00	74.00	74.00	69.50	73.25	81.00	73.50	73.31	67.50	78.50
$\sigma$	4.113	7.724	1.633	5.416	7.832	1.915	2.872	4.163	3.416	7.003	7.724	4.435

**Tabla 5: Rendimiento de los mejores individuos para todas las ejecuciones para la clase Persona de GRAZ-01.**

GRAZ-01 clase Persona												
	Corr 1			Corr 2			Corr 3			Corr 4		
Doblez	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	68.00	78.00	64.00	73.00	72.00	74.00	81.00	68.00	72.00	78.00	72.67	74.00
2	73.00	68.00	70.00	<b>78.00</b>	<b>66.00</b>	<b>76.00</b>	84.00	74.00	76.00	74.00	64.00	74.00
3	76.00	72.00	70.00	78.00	74.00	71.33	75.00	74.00	68.00	100.00	68.25	72.00
4	70.00	74.00	64.00	78.00	72.00	70.00	74.00	72.00	76.00	70.00	78.00	68.00
Promedio	71.75	73.00	67.00	76.75	71.00	72.83	78.50	72.00	73.00	80.50	70.73	72.00
$\sigma$	3.500	4.163	3.464	2.500	3.464	2.687	4.796	2.828	3.830	13.404	6.001	2.828

	Corr 5			Corr 6			Corr 7			Corr 8		
Doblez	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	72.00	72.00	72.00	77.00	67.60	68.00	80.00	76.00	74.00	64.00	64.50	70.00
2	74.00	70.00	72.00	73.00	78.67	74.00	88.00	70.00	74.00	75.00	76.00	66.00
3	72.00	70.00	72.00	76.38	64.00	70.00	71.00	72.00	66.00	77.00	70.00	70.00
4	81.00	76.00	66.00	75.00	74.00	68.00	79.00	70.00	66.00	76.00	72.00	74.00
Promedio	74.75	72.00	70.50	75.34	71.07	70.00	79.50	72.00	70.00	77.25	74.00	71.00
$\sigma$	4.272	2.828	3.000	1.772	6.540	2.828	6.952	2.828	4.619	2.630	3.651	3.830

**Tabla 6: Rendimiento de los mejores individuos para todas las ejecuciones para la clase Bicicleta de GRAZ-02.**

GRAZ-02 clase Bicicleta												
	Corr 1			Corr 2			Corr 3			Corr 4		
Doblez	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	86.67	85.33	77.33	77.33	76.00	80.00	77.33	81.33	78.67	78.67	77.33	81.33
2	88.00	82.67	80.00	82.67	80.00	80.00	84.00	80.00	80.00	82.00	78.67	84.00
3	84.67	82.67	77.33	79.33	78.67	81.33	82.00	80.00	77.33	76.00	80.00	80.00
4	91.33	78.67	80.00	80.67	78.67	82.67	82.00	84.00	80.00	89.33	81.51	84.00
Promedio	87.67	82.33	78.67	80.00	78.33	81.00	81.33	81.33	79.00	81.50	79.38	82.33
$\sigma$	2.802	2.749	1.540	2.244	1.678	1.277	2.828	1.886	1.277	5.770	1.790	2.000

	Corr 5			Corr 6			Corr 7			Corr 8		
Doblez	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	84.67	80.00	78.67	75.33	81.33	80.00	78.00	76.00	81.33	87.33	77.33	81.33
2	83.33	84.00	82.67	86.67	80.00	77.33	82.67	80.00	81.33	80.67	78.67	78.67
3	78.00	76.00	81.33	76.67	72.00	81.33	80.67	76.00	82.67	80.00	80.00	82.67
4	<b>89.33</b>	<b>82.67</b>	<b>86.67</b>	82.67	86.67	84.00	88.00	81.33	85.33	78.00	82.67	80.00
Promedio	83.83	80.67	82.33	80.33	80.00	80.67	82.33	78.33	82.67	81.50	79.67	80.67
$\sigma$	4.663	3.528	3.333	5.292	6.061	2.776	4.234	2.749	1.886	4.051	2.277	1.721

**Tabla 7: Rendimiento de los mejores individuos para todas las ejecuciones para la clase Persona de GRAZ-02.**

GRAZ-02 clase Persona												
	Corr 1			Corr 2			Corr 3			Corr 4		
Doblez	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	98.00	88.00	85.33	90.00	84.00	88.00	85.33	80.00	86.67	83.33	78.67	84.00
2	91.33	82.67	84.00	87.33	81.33	84.00	82.00	88.00	84.00	98.00	82.67	82.67
3	86.67	88.00	84.00	80.67	72.00	82.67	99.33	88.00	86.67	97.33	89.33	86.67
4	86.00	84.00	84.00	96.00	81.56	86.67	84.67	89.33	84.00	98.00	85.33	82.67
Promedio	90.50	85.67	84.33	88.50	79.72	85.33	87.83	86.33	85.33	94.17	84.00	84.00
$\sigma$	5.534	2.749	0.667	6.357	5.288	2.434	7.801	4.269	1.540	7.229	4.489	1.886

	Corr 5			Corr 6			Corr 7			Corr 8		
Doblez	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	83.33	78.67	84.00	92.67	86.67	84.00	<b>87.33</b>	<b>89.33</b>	<b>90.67</b>	98.00	85.33	85.33
2	98.00	82.67	82.67	89.33	84.00	80.00	99.33	85.64	89.33	80.00	77.33	82.67
3	97.33	89.33	86.67	88.00	86.67	82.67	85.33	88.00	89.33	89.33	82.67	84.00
4	98.00	85.33	82.67	94.67	89.33	85.33	92.00	90.67	85.33	100.00	82.67	86.67
Promedio	94.17	84.00	84.00	91.17	86.67	83.00	91.00	88.41	88.67	91.83	82.00	84.67
$\sigma$	7.229	4.489	1.886	3.049	2.177	2.277	6.218	2.142	2.309	9.147	3.355	1.721

**Tabla 8: Rendimiento de los mejores individuos para todas las ejecuciones para la clase Carros de GRAZ-02.**

GRAZ-02 clase Carros												
Doblez	Corr 1			Corr 2			Corr 3			Corr 4		
	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	78.00	84.00	81.33	84.67	84.00	84.00	78.00	78.67	78.67	86.00	77.33	80.00
2	86.00	77.33	84.00	90.67	86.67	84.00	86.67	81.33	80.00	88.67	78.67	76.00
3	<b>80.00</b>	<b>81.33</b>	<b>85.33</b>	83.33	84.00	82.67	86.67	78.67	81.33	80.67	72.00	80.00
4	<b>83.33</b>	<b>81.33</b>	<b>85.33</b>	83.33	84.00	84.00	87.33	82.67	80.00	83.33	76.00	82.00
Promedio	81.83	81.00	84.00	85.50	84.67	83.67	84.67	80.33	80.00	84.67	76.00	79.50
$\sigma$	3.543	2.749	1.886	3.501	1.333	0.667	4.456	2.000	1.089	3.443	2.880	2.517

Doblez	Corr 5			Corr 6			Corr 7			Corr 8		
	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba	Ent.	Val.	Prueba
1	82.00	85.33	81.33	84.67	78.67	84.00	76.67	80.00	77.33	86.67	74.67	77.33
2	82.67	84.00	84.00	84.00	74.67	80.00	84.00	82.67	84.00	80.00	77.33	77.33
3	82.00	80.00	80.00	78.00	81.33	84.00	78.67	81.33	76.00	78.67	81.33	78.67
4	82.67	86.67	84.00	80.67	80.00	82.67	84.00	78.67	82.67	83.33	81.33	80.00
Promedio	82.33	84.00	82.33	81.83	78.67	82.67	80.83	80.67	80.00	82.17	78.67	78.33
$\sigma$	0.385	2.880	2.000	3.097	2.880	1.886	3.747	1.721	3.925	3.585	3.266	1.277

Después de realizar las corridas evolutivas, los mejores individuos de caja ejecución fueron seleccionados para realizar pruebas adicionales; resultando en 32 individuos por clase, es decir, 160 soluciones en total. Cabe mencionar que en este caso la selección del mejor individuo se hizo en base a su rendimiento en la etapa de pruebas durante el proceso de búsqueda. Entonces, cada uno de los individuos seleccionados fue utilizado para clasificar otro conjunto de pruebas de 75 imágenes por cada clase. El rendimiento de clasificación del modelo AVCMO en cada una de las clases de GRAZ se muestra en la Tabla 9 junto con sistemas del estado-del-arte. La comparación entre los sistemas se realiza a través de la medida EER. El modelo propuesto en esta tesis muestra resultados similares al mejor algoritmo *EBIM* para la clase Bicicleta de GRAZ-01, mientras que supera en calidad de clasificación a todos los sistemas en las tres clases de GRAZ-02. También se muestra el rendimiento promedio de las 32 soluciones seleccionadas para cada una de las clases, el intervalo de confianza presente en la tabla, se calculó con un nivel de confianza del 99 %.

Además de los resultados del rendimiento de clasificación presentados en la tabla anterior, las siguientes figuras muestran las curvas de característica operativa del receptor (ROC, por sus siglas en inglés *receiver operating characteristic*). Estas gráficas muestran la relación entre la razón de verdaderos positivos (o sensibilidad), y la razón de falso positivos (1-especificidad) en un clasificador binario mientras se varia el umbral de discrimi-

**Tabla 9: Comparación del rendimiento del modelo AVCMO en clasificar las imágenes de la base de datos de GRAZ. Los valores representan el porcentaje de clasificación para cada clase.**

Modelo	GRAZ-01		GRAZ-02		
	Bike	Person	Bike	Person	Cars
Moment Invariants + Aff. Inv (Opelt <i>et al.</i> , 2006)	73.5	63.0	72.5	81.0	67.0
Basic Moments + Aff. Inv (Opelt <i>et al.</i> , 2006)			76.5	77.2	70.2
DoG + SIFT (Opelt <i>et al.</i> , 2006)	78.0	76.5	76.4	70.0	68.9
SM + Intensity distributions (Opelt <i>et al.</i> , 2006)	83.5	56.5	74.0	74.1	56.5
Mutch et al. (Mutch y Lowe, 2008)			80.5	81.7	70.1
EBIM (Huang <i>et al.</i> , 2011)	<b>84.1</b>	<b>86.0</b>	80.8	83.2	72.2
HMAX-GA (Ghodrati <i>et al.</i> , 2012)	80.2	84.0	82.6	82.3	75.6
AVCMO Mejor	84.0	76.0	<b>86.67</b>	<b>90.67</b>	<b>85.33</b>
AVCMO Promedio	74.25 ± 2.09	70.79 ± 1.59	80.92 ± 1.06	84.92 ± 1.06	81.31 ± 1.25

nación. El análisis con las curvas ROC brinda herramientas para comprar el rendimiento de un clasificador sin importar las distribución de las clases. De esta manera, las Figuras 30, 31, 32, 33 y 34 muestran la comparación del modelo AVCMO con otros del estado del arte. En las gráficas se observa que en promedio el modelo AVCMO logra resultados de clasificación similares a los del estado del arte en ambas clases de GRAZ-01. Mientras que el individuo que logra el mejor rendimiento en clasificación del conjunto de pruebas, alcanza resultados similares al sistema EBIM en la clase Bicicleta.

Por otra parte, el rendimiento promedio de los individuos evolucionados es mejor al compararlo con siete sistemas del estado-del-arte, superándolos en la clasificación de las tres clases de GRAZ-02. Incluso, el mejor individuo encontrado alcanza porcentajes de clasificación hasta 10% mejor que los otros métodos, enriqueciendo considerablemente los resultados del estado del arte. Cabe notar que el método propuesto supera los sistemas estudiados en cuatro de las cinco clases de imágenes utilizadas en estos experimentos. Estos resultados son similares a los de los métodos del estado-del-arte, como podemos observar en la literatura (Mutch y Lowe, 2008; Ghodrati *et al.*, 2012; Huang *et al.*, 2011; Opelt *et al.*, 2006), y podemos concluir que la programación de cerebros es un viable como técnica de optimización para problemas complejos de clasificación, como la tarea estudiada en este trabajo, dado que el modelo logra resultados que alcanzan, y en algunos casos superan, a los métodos tradicionales.

La Figura 35 muestra los individuos correspondientes a la última generación de cada una de las 32 realizaciones del proceso evolutivo para la clase Bicicleta de GRAZ-01. Las soluciones están representadas por sus valores de aptitud, ( $1/EE R$ ) contra ( $1/\text{Average } F\text{-measure}$ ). La Figura 36 presenta los valores de aptitud para la clase Perso-

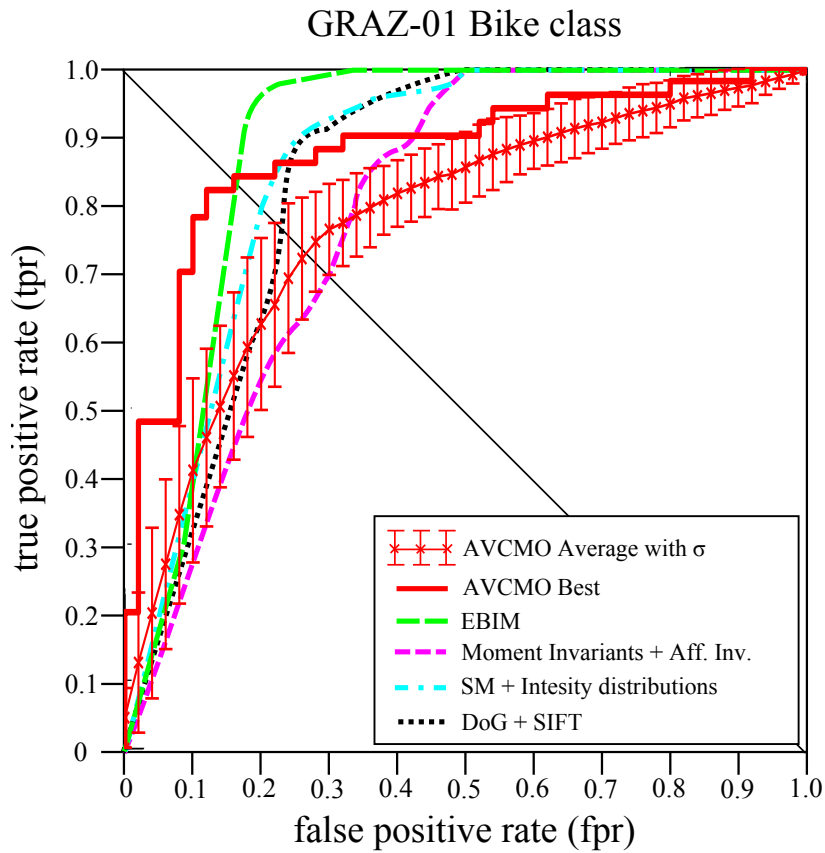


Figura 30: Comparación usando las curvas ROC para la clase Bicicleta de GRAZ-01.

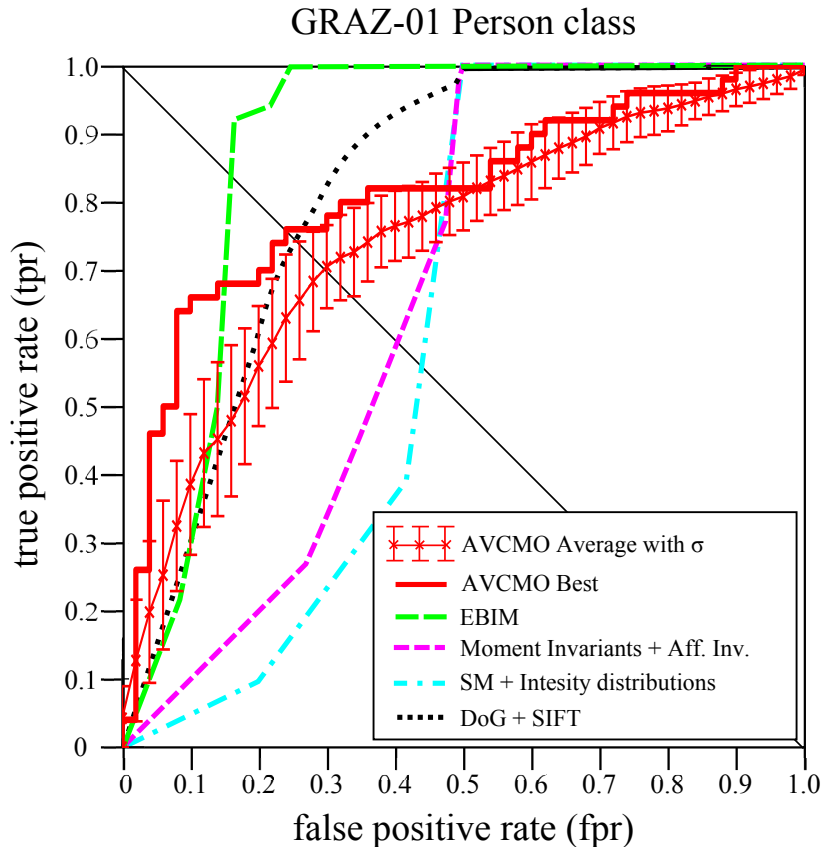


Figura 31: Comparación usando las curvas ROC para la clase Persona de GRAZ-01.

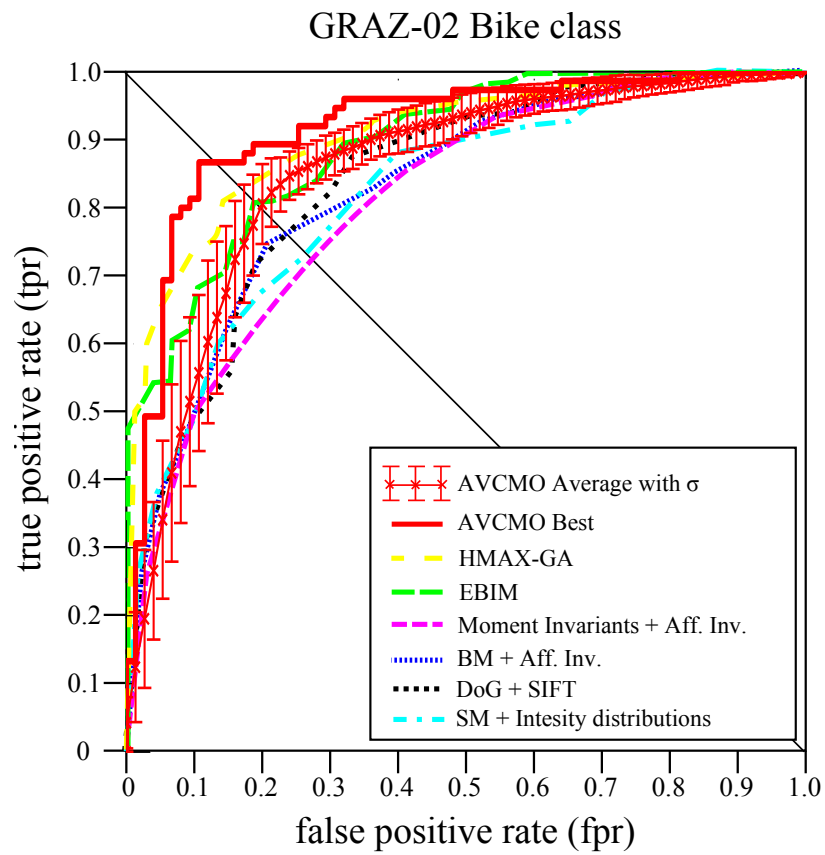


Figura 32: Comparación usando las curvas ROC para la clase Bicicleta de GRAZ-02.

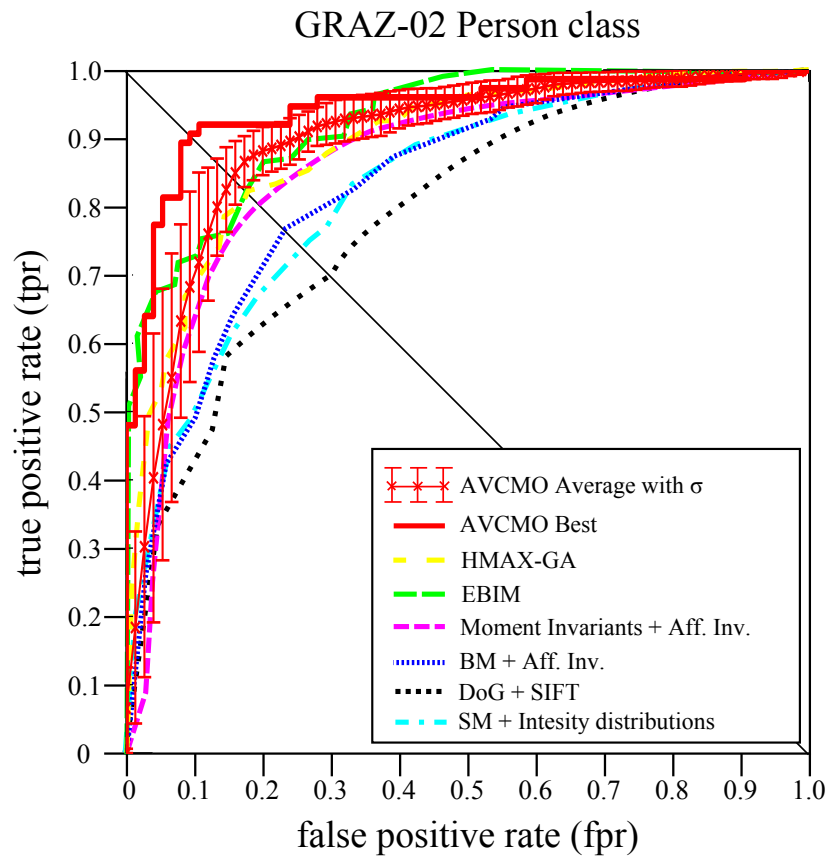
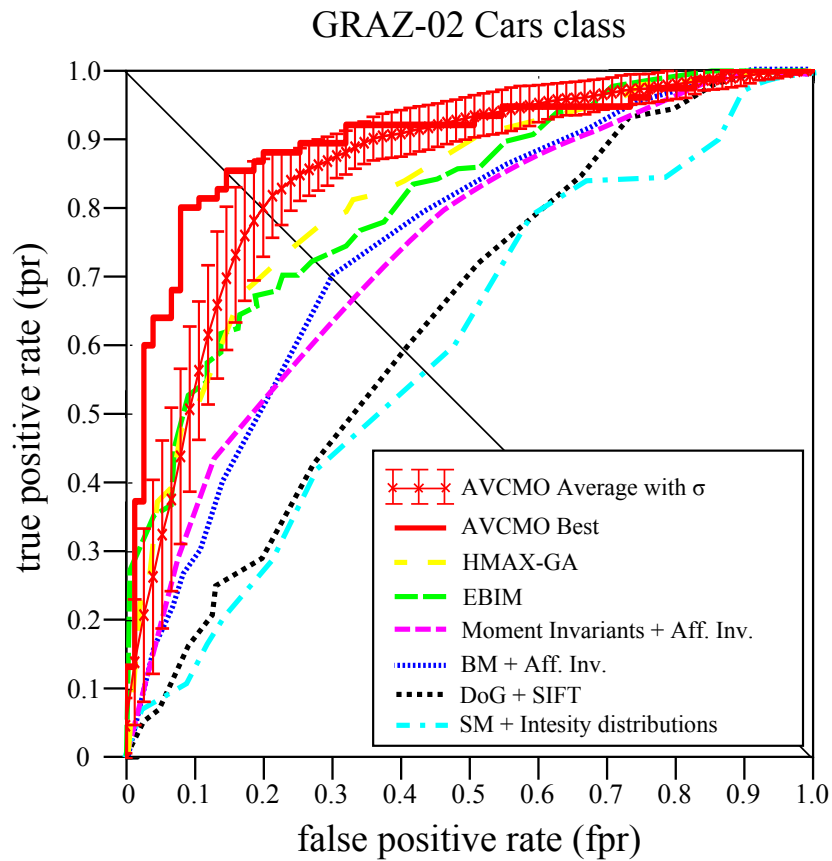


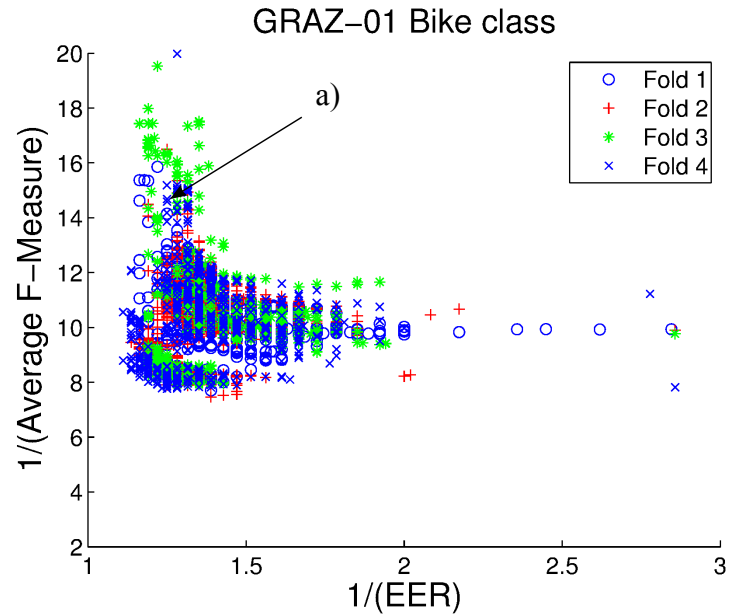
Figura 33: Comparación usando las curvas ROC para la clase Persona de GRAZ-02.



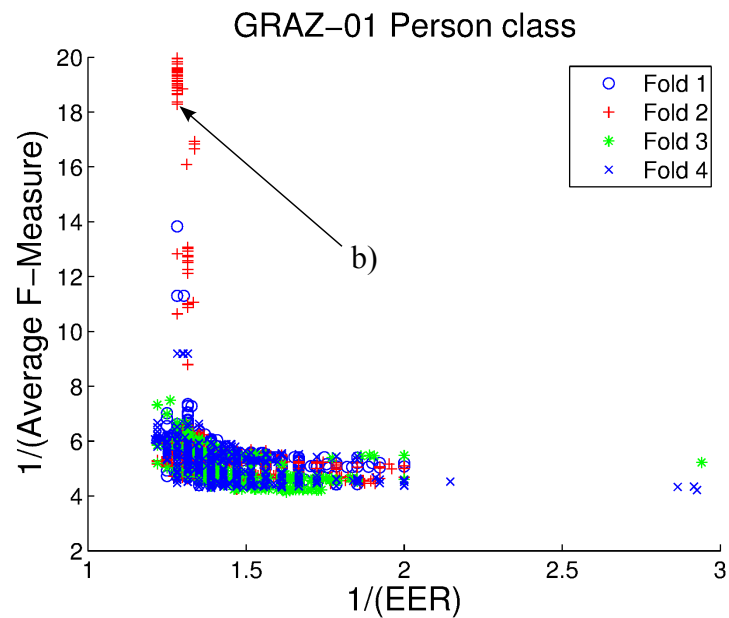
**Figura 34: Comparación usando las curvas ROC para la clase Carros de GRAZ-02.**

na GRAZ-01. De la misma manera, la Figura 37, 38 y 39 muestra los valores correspondientes a las clases Bicicleta, Persona y Carros respectivamente; estas clases corresponden al conjunto GRAZ-02. Estas gráficas muestran los frentes que se logran después del proceso evolutivo. Los elementos que se encuentran más cerca del origen presentan un buen rendimiento con respecto a las dos funciones objetivo. Sin embargo, existen soluciones que presentan un buen rendimiento con respecto a la clasificación, mientras que su calidad en el proceso de localización se reduce; y viceversa, hay soluciones que presentan mejores resultados en la localización, pero su rendimiento de clasificación desciende.

Además, las Figuras 40, 41, 42, 43 y 44 muestran la convergencia de ambas funciones objetivos a lo largo de las 30 generaciones en el proceso evolutivo; los resultados corresponden a las clases de los dos sets GRAZ-01 y GRAZ-02. Partiendo de los resultados que se muestran en estas gráficas, resulta interesante buscar una metodología distinta para la creación de las soluciones, una posibilidad sería explorar los resultados de un proceso de búsqueda aleatoria.

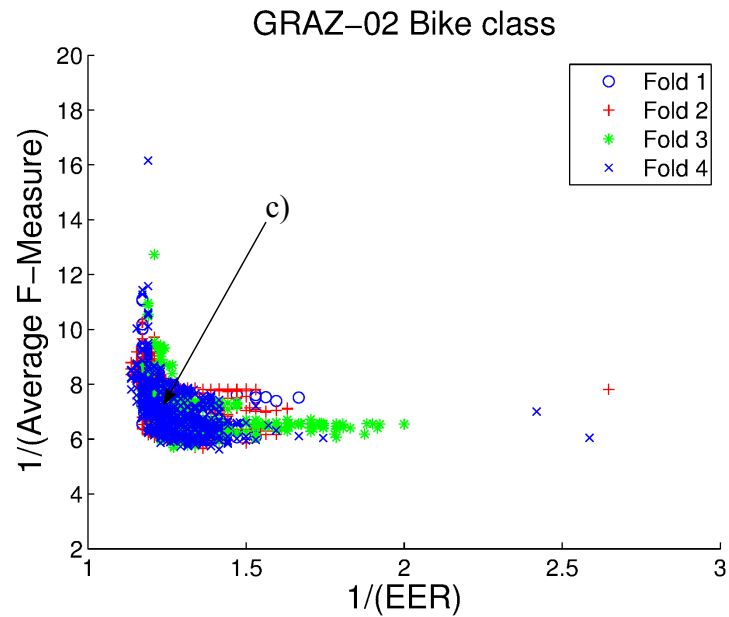


**Figura 35:** Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Bicicleta de GRAZ-01. El individuo señalado corresponde a la solución presente en la Tabla 10.

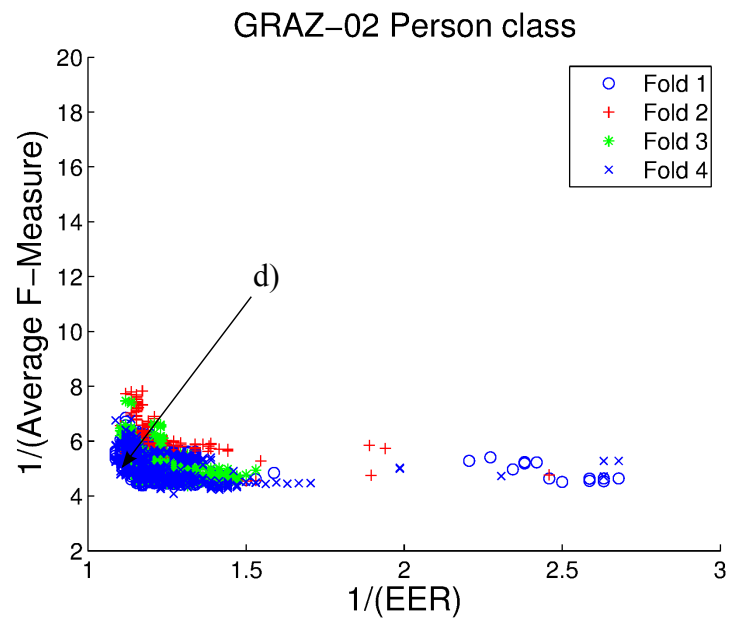


**Figura 36:** Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Persona de GRAZ-01. El individuo señalado corresponde a la solución presente en la Tabla 10.

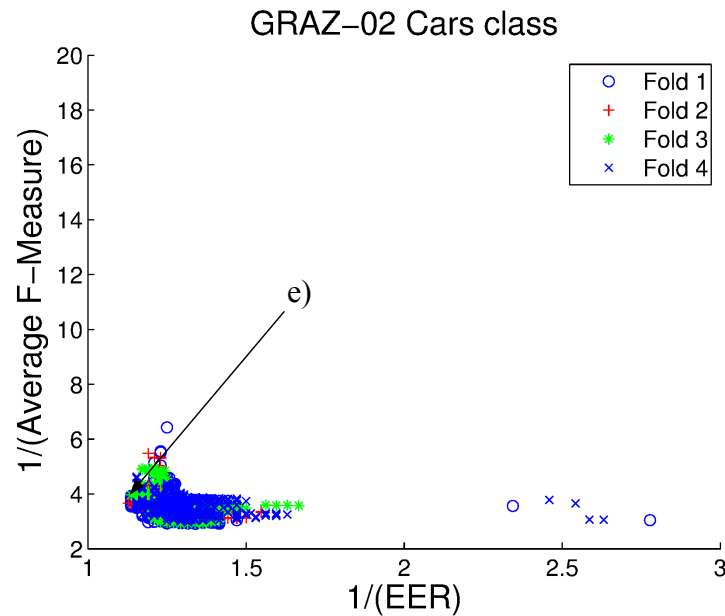




**Figura 37:** Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Bicicleta de GRAZ-02. El individuo señalado corresponde a la solución presente en la Tabla 10.



**Figura 38:** Soluciones de la última generación de las ocho corridas de los cuatro dobles para la clase Persona de GRAZ-02. El individuo señalado corresponde a la solución presente en la Tabla 10.



**Figura 39: Soluciones de la última generación de las ocho corridas de los cuatro dobleces para la clase Carros de GRAZ-02. El individuo señalado corresponde a la solución presente en la Tabla 10.**

Una vez seleccionados los mejores individuos de cada proceso evolutivo, resulta interesante analizar los operadores que los componen. Así, se realizó un estudio para encontrar qué dimensión de características tenía mayor impacto en encontrar información prominente sobre los objetos de las diferentes clases. Por ende, se realizó un análisis de la frecuencia de uso sobre las funciones y terminales presentes en la Tabla 2. Las Figuras 45, 46, 47, 48, 49, 50, 51 y 52 presentan cuantas veces se implementa cada una de las funciones y terminales en los mejores individuos de la última generación de cada corrida evolutiva. Las gráficas en las Figuras 45(a) y 45(b) muestran la frecuencia de uso para la dimensión de color. Las Figuras 46(a) y 46(b) presentan los resultados para la dimensión de orientación. Además, las Figuras 51(a) y 47(b) exhiben los valores para la dimensión de forma. Y finalmente, las Figuras 48 y 52 expone la frecuencia de uso para la construcción de los operadores de lo mapas mentales.

De esta manera, los resultados para el set GRAZ-01 muestran que las funciones más utilizadas para la dimensión de color fueron *square*, *multiplication* y *threshold* en la clase Bicicleta, y *square root* y *threshold* para Persona; mientras que las terminales más frecuentes fueron la oponencia de color *blue – yellow* para Bicicleta y los canales *red* y *hue* para Persona. En el caso de la dimensión de orientación, las funciones predominantes fueron  $D_x$  y  $D_y$  para ambas clases, y las terminales más utilizadas fueron los canales *yellow* y

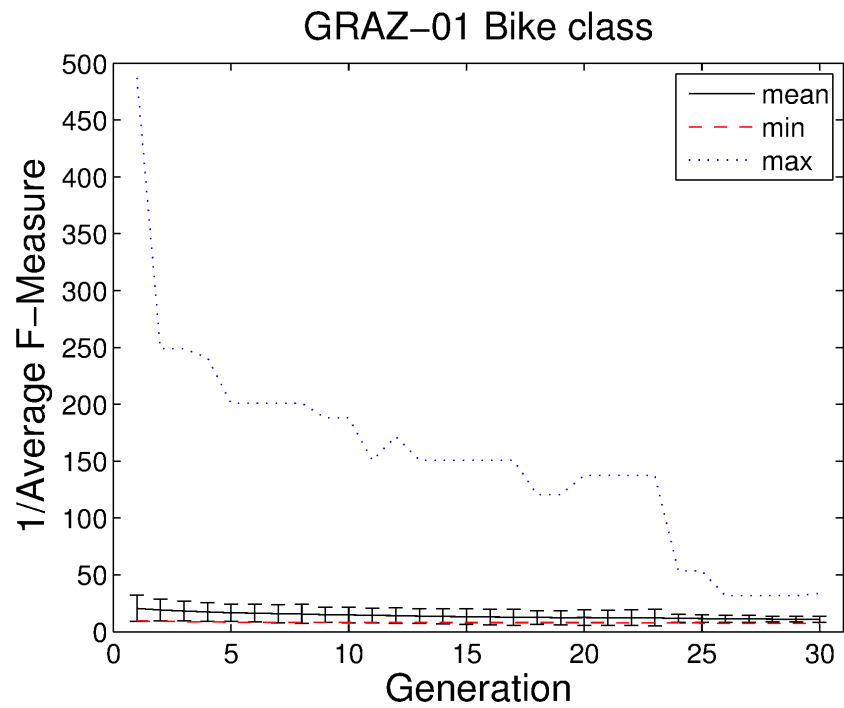
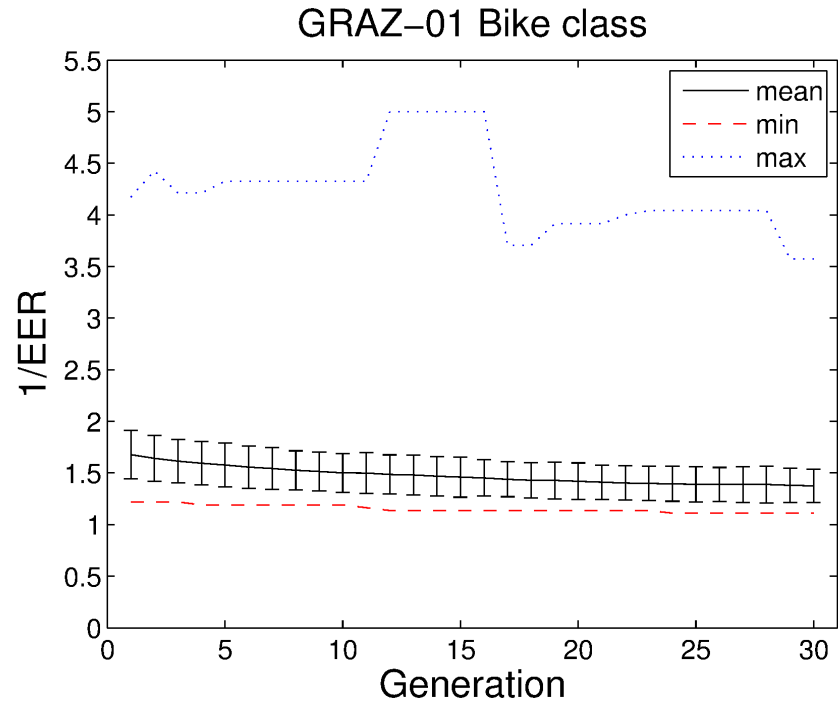


Figura 40: Gráficas de convergencia del algoritmo de BP para la clase Bicicleta de GRAZ-01. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobles.

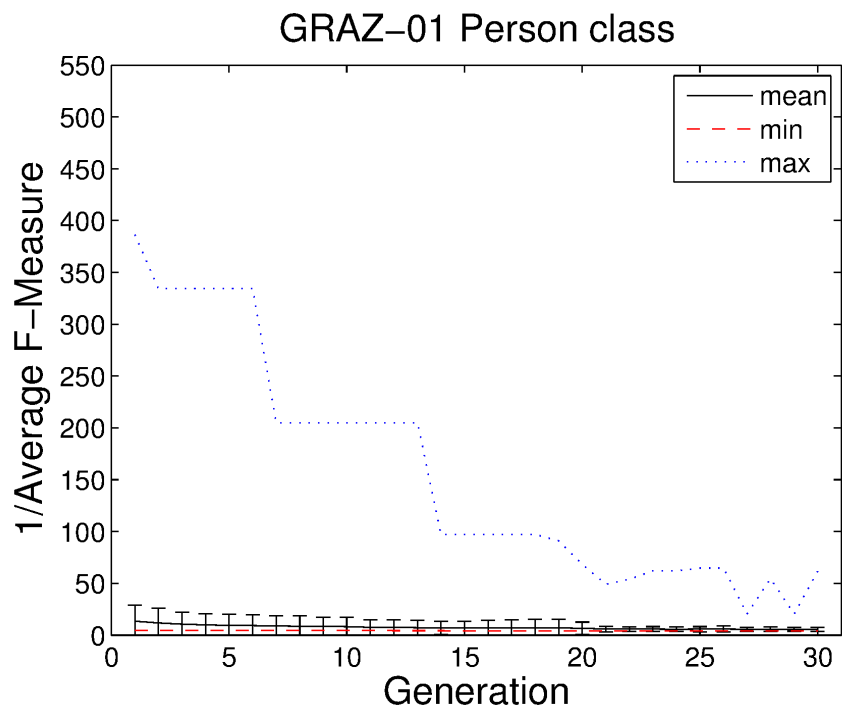
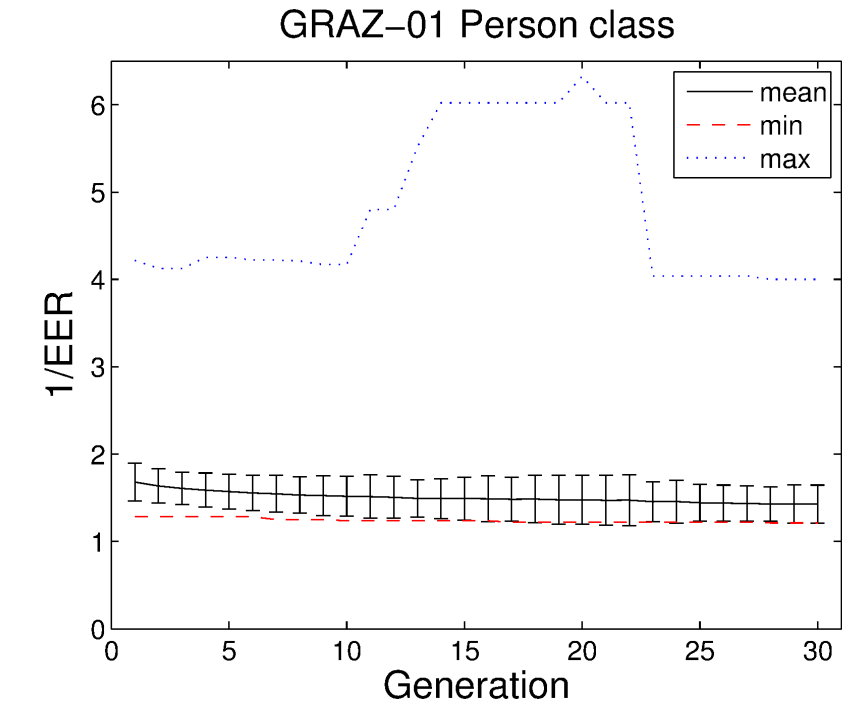


Figura 41: Gráficas de convergencia del algoritmo de BP para la clase Persona de GRAZ-01. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobleces.

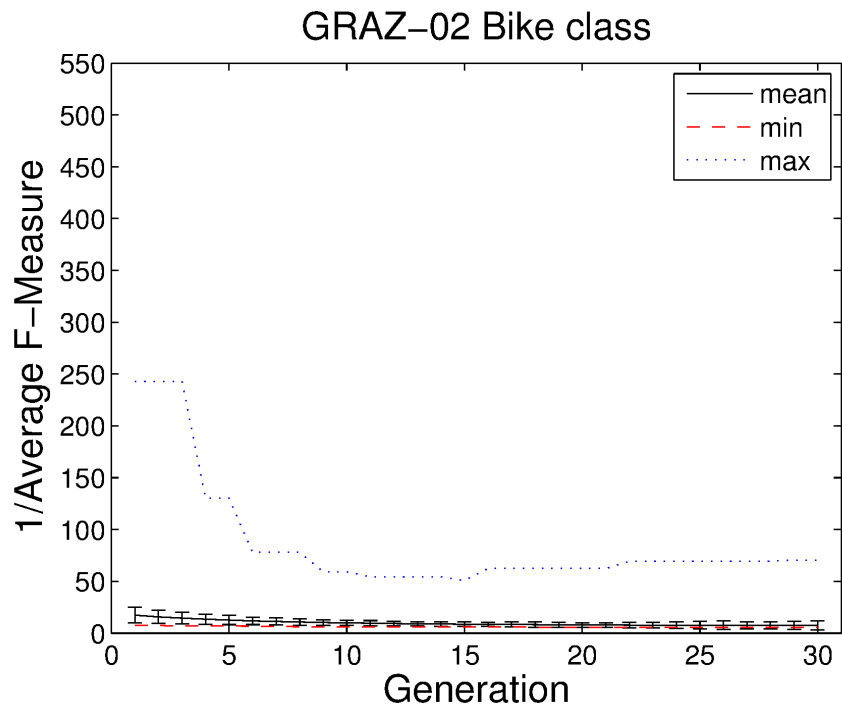
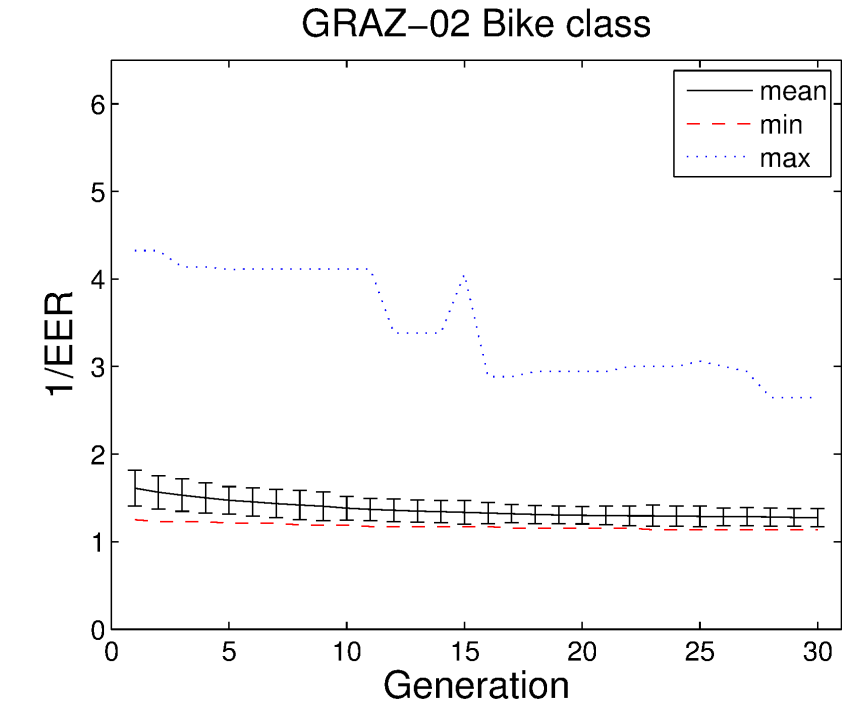


Figura 42: Gráficas de convergencia del algoritmo de BP para la clase Bicicleta de GRAZ-02. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobles.

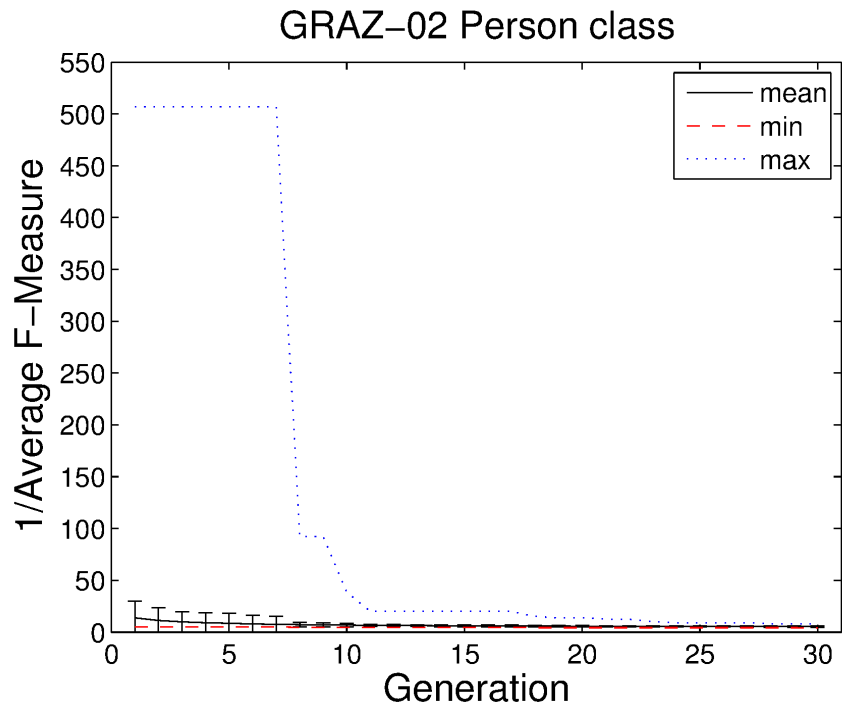
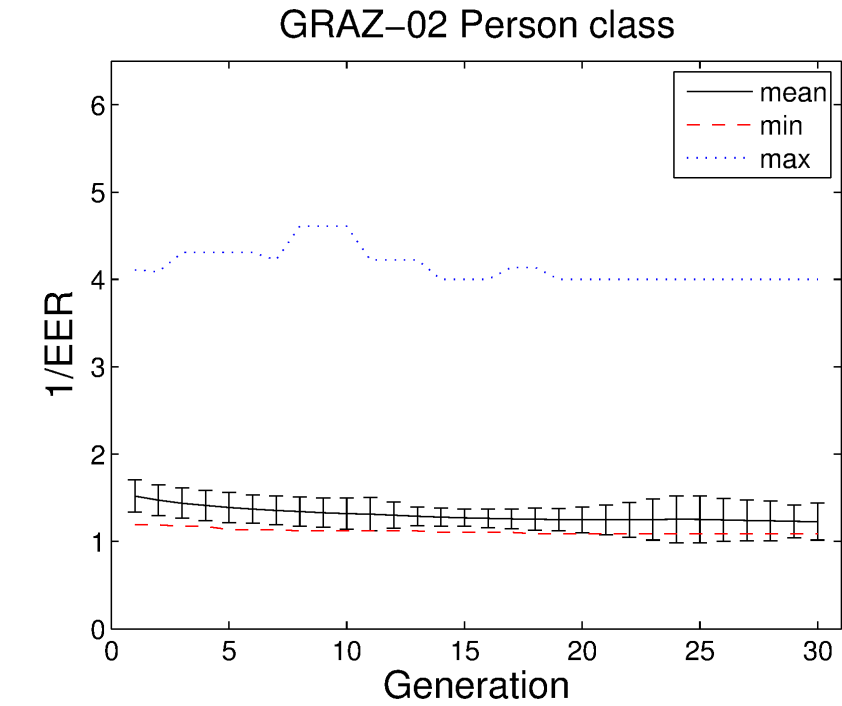


Figura 43: Gráficas de convergencia del algoritmo de BP para la clase Persona de GRAZ-02. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobleces.

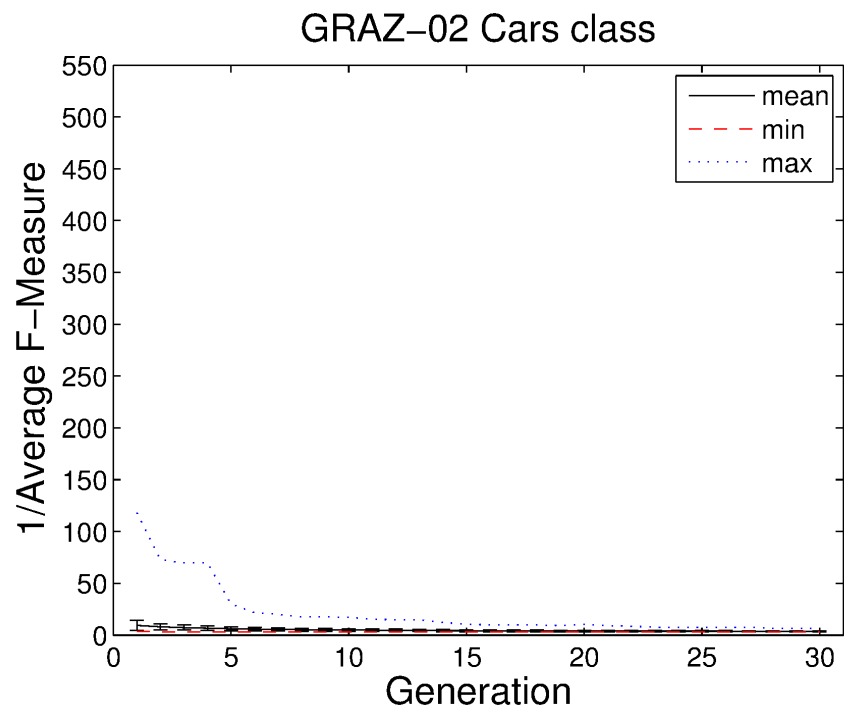
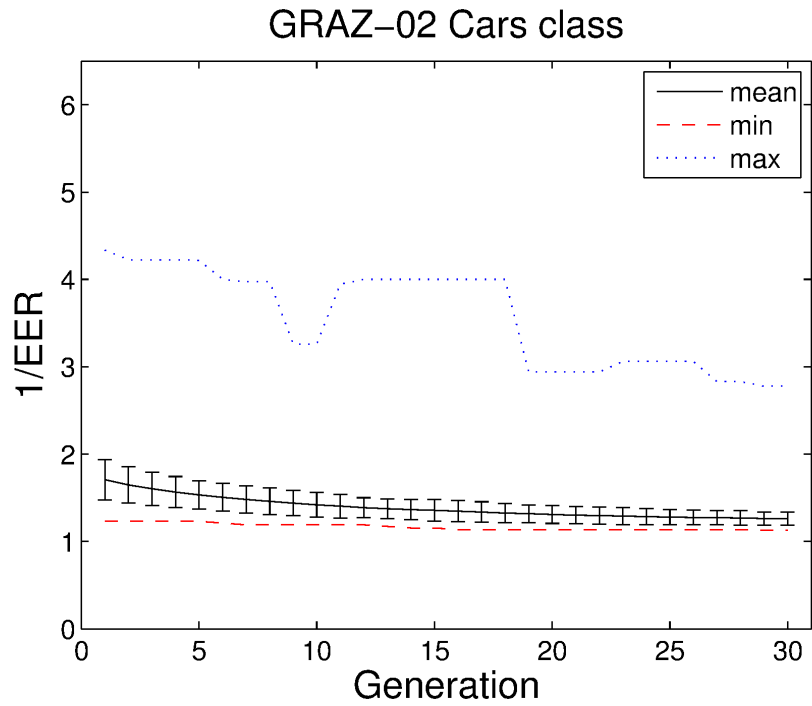


Figura 44: Gráficas de convergencia del algoritmo de BP para la clase Carros de GRAZ-02. Se muestran la media y desviación estándar de las dos funciones objetivo durante las 30 generaciones de las ocho corridas de los cuatro dobleces.

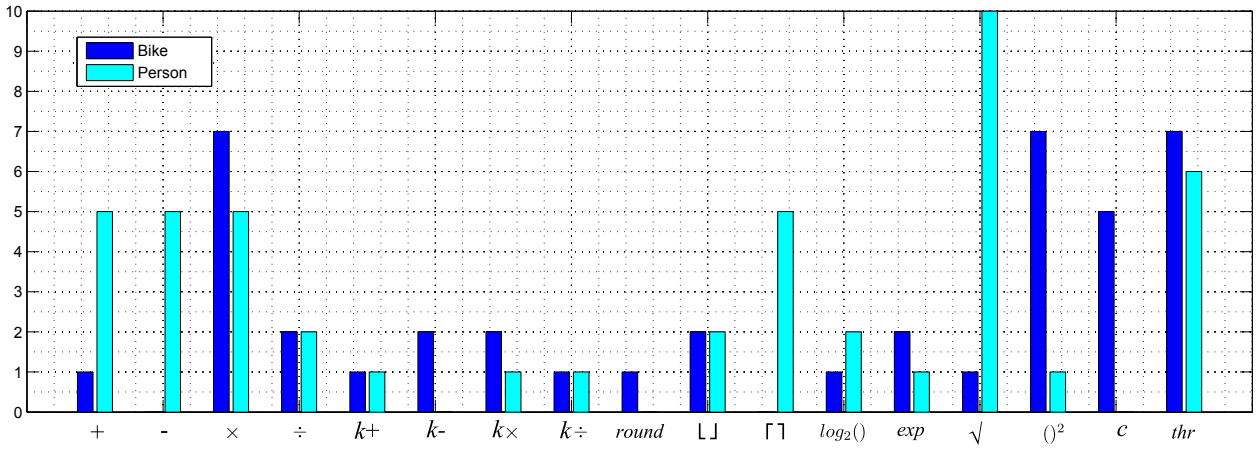
*magenta* para Bicicleta y *red* para Persona. La dimensión de forma se caracterizó por la función *dilate* en Bicicletas y *multiplication* y *division* en Personas; las terminales más comunes fueron *yellow* en bicicletas, y *red*, *black* y *hue* para Persona. Finalmente, las funciones más frecuentes para la creación de los mapas mentales fueron  $D_x$  y  $D_y$ .

Los mejores individuos para las clases de GRAZ-02 se componen principalmente de las operaciones *multiplication*, *exponential* y *threshold* en la dimensión de color para las tres clases; mientras que la terminal más frecuente fue el canal *hue*. En la dimensión de orientación las funciones más usadas fueron  $D_x$  y  $D_y$  en los tres casos, pero las terminales más usadas fueron *magenta* y *saturation* para Bicicleta, *hue* y *red* para Persona, y *grayscaleimage* y *black* para Carros. En la dimensión de forma predominaron las funciones *Bottom – Hat* y *dilation* para Bicicleta, *multiplication* y *division*, y *dilation* en los Carros; en cuanto a las terminales, el canal *magenta* dominó en las clases Bicicleta y Carros, mientras que *black* y *hu* fueron más frecuentes en Persona. Al igual que para el conjunto GRAZ-01, las funciones más implementadas para la creación de los mapas mentales son las derivadas  $D_x$  y  $D_y$  para todas las clases.

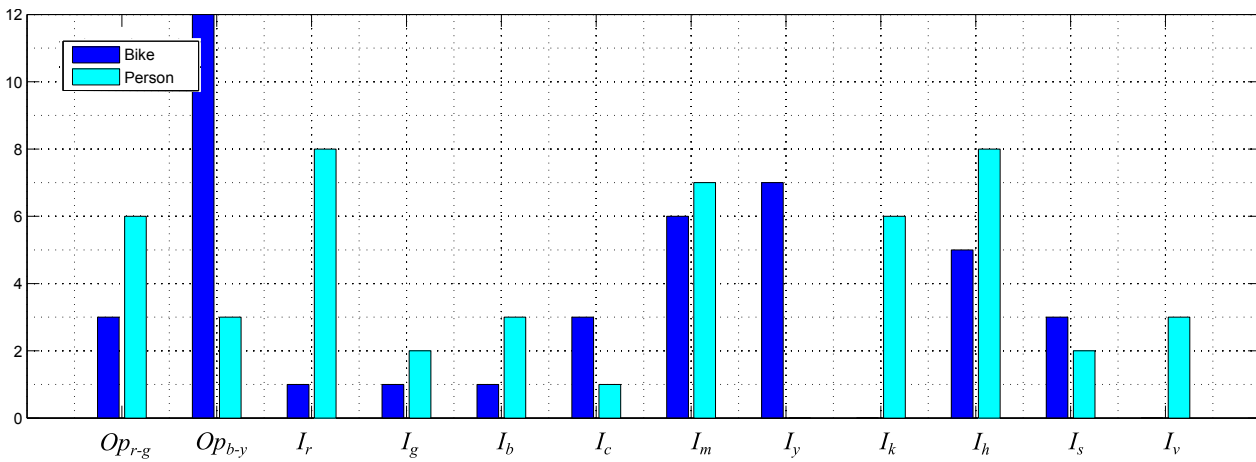
En base a estos resultados de frecuencia de uso, se podrían reducir los conjuntos de funciones y terminales con el objetivo de lograr mejores soluciones. Si se utilizan solo las funciones que han mostrado buenos resultados, se puede tener una convergencia más rápida del algoritmo evolutivo; así como lograr mejores soluciones, al realizar la búsqueda en un espacio de soluciones más pequeño.

Una de las problemáticas de un enfoque multi-objetivo es seleccionar la mejor solución. En este trabajo los mejores individuos fueron seleccionados en base a su desempeño en clasificación, la Tabla 10 muestra los mejores individuos encontrados después de las 32 ejecuciones del proceso evolutivo. Además, se muestran algunas imágenes de ejemplo del funcionamiento del sistema propuesto; la región marcada corresponde a los puntos que fueron utilizados para la creación del vector descriptor. Los individuos presentes en esta tabla brindan una idea general de la estructura de las soluciones que se pueden crear en base al modelo propuesto, además de ejemplificar el nivel de complejidad de las operaciones que se logran a través del paradigma de BP. Asimismo, estas soluciones son buenos candidatos para un trabajo futuro, enfocado en hacer un estudio



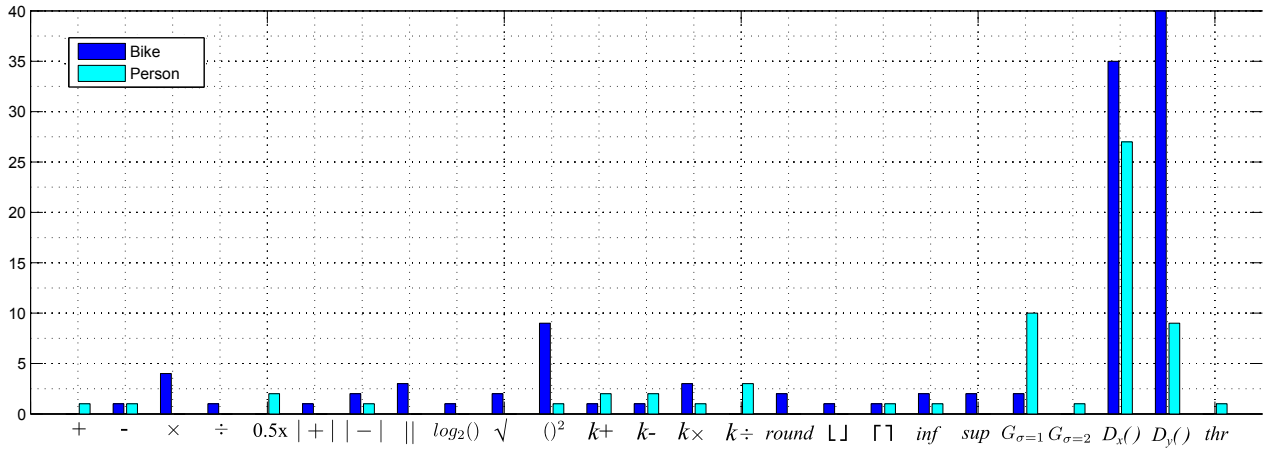


(a)

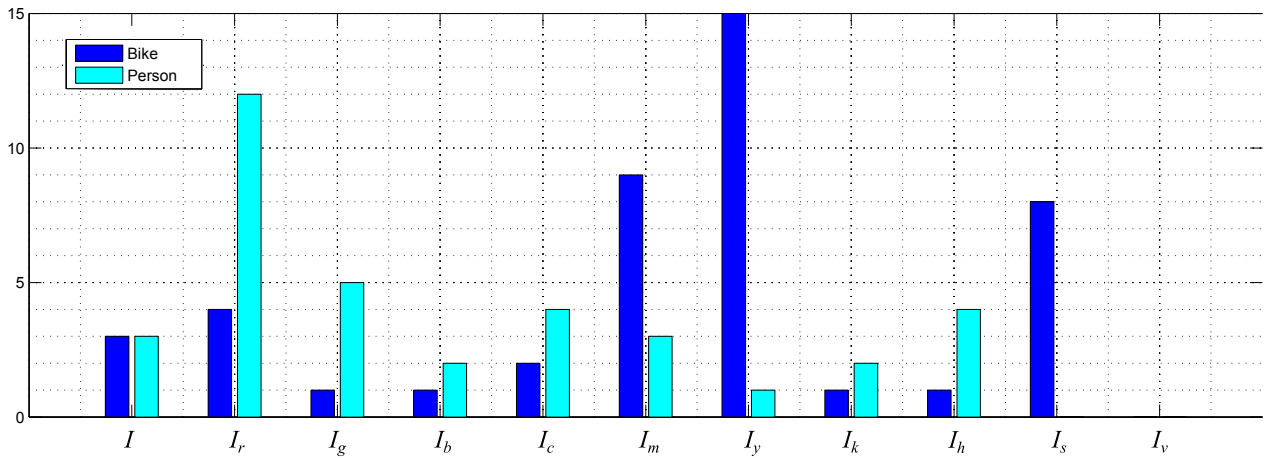


(b)

Figura 45: Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de color, utilizadas para crear las soluciones para las clases de GRAZ-01.



(a)



(b)

Figura 46: Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de orientación, utilizadas para crear las soluciones para las clases de GRAZ-01.

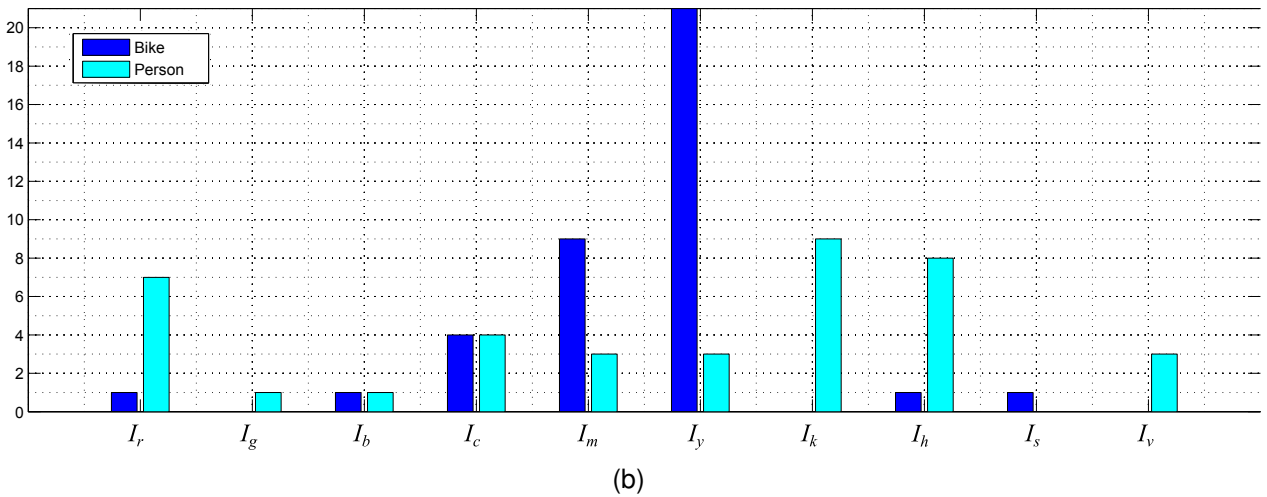
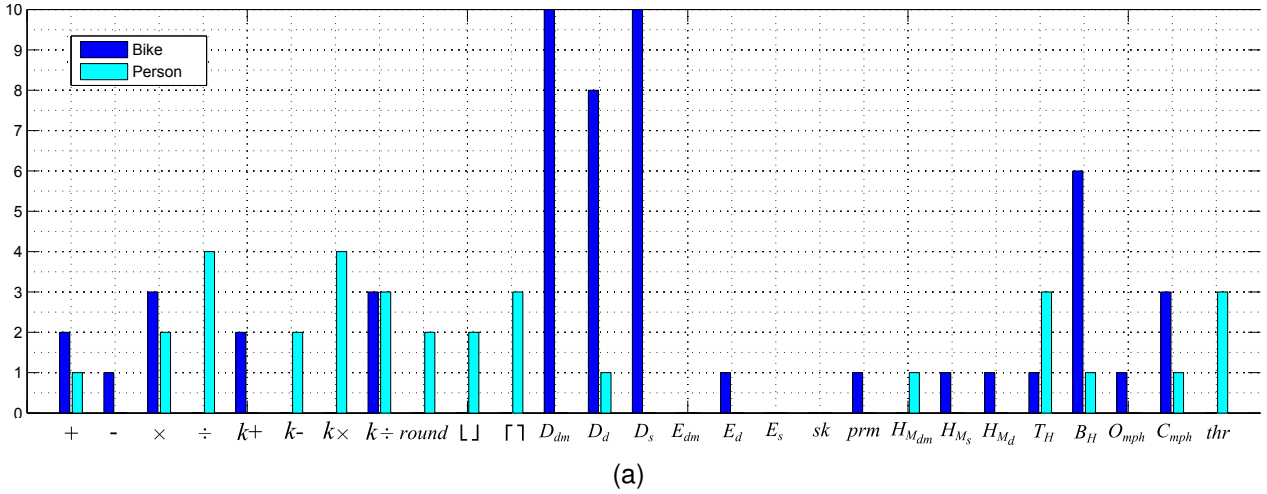


Figura 47: Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de forma, utilizadas para crear las soluciones para las clases de GRAZ-01.

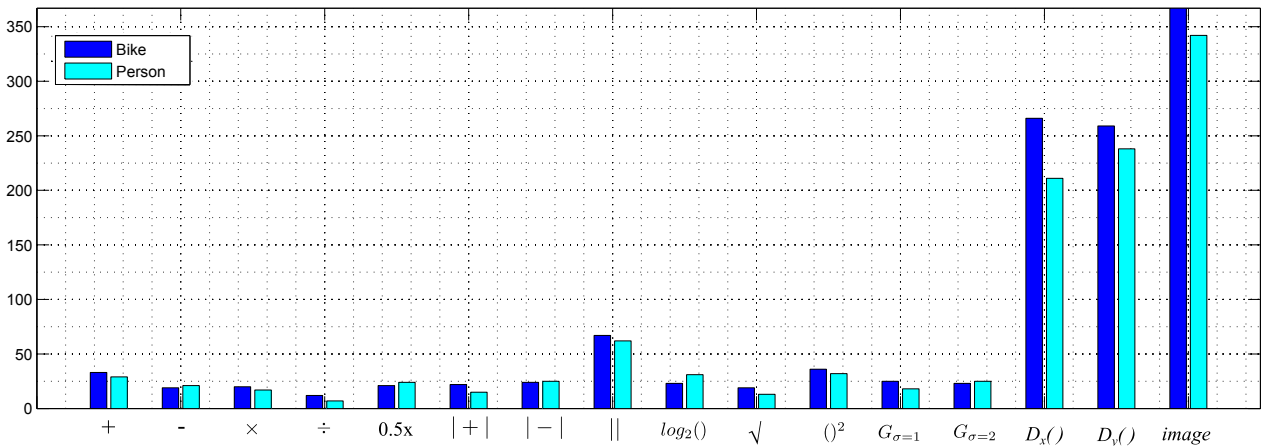
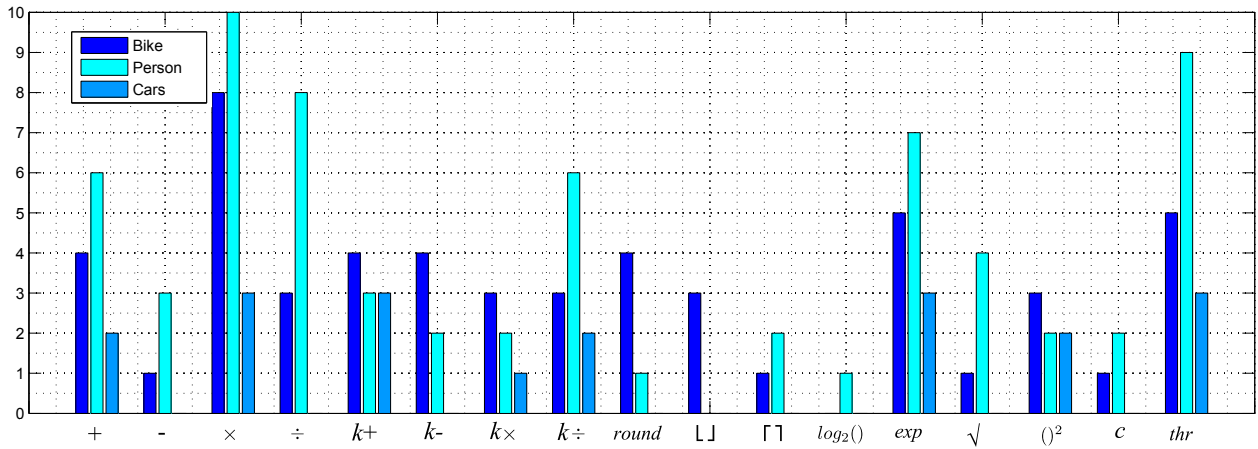
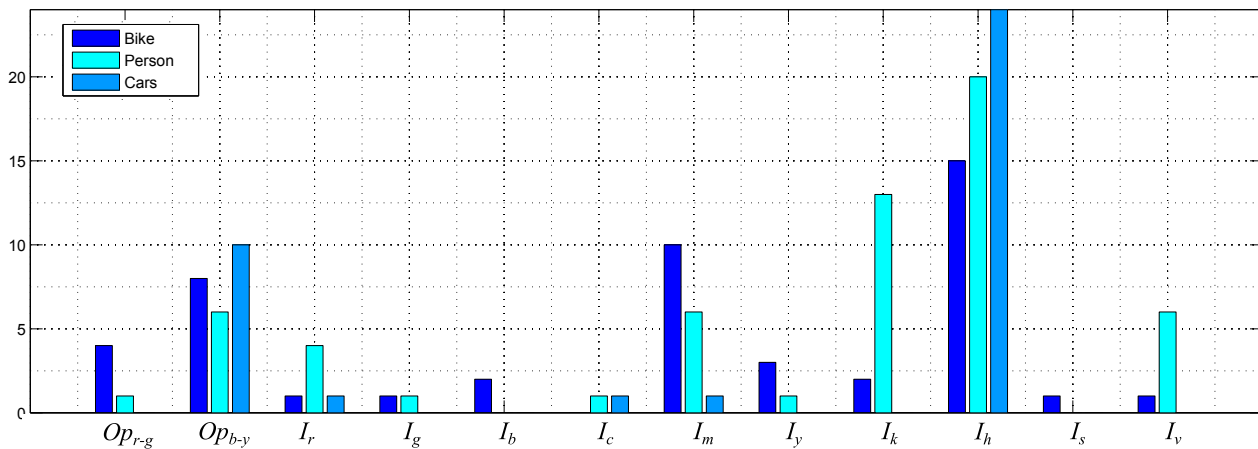


Figura 48: Gráficas de la frecuencia de uso de las funciones y terminales para los mapas mentales, utilizadas para crear las soluciones para las clases de GRAZ-01.

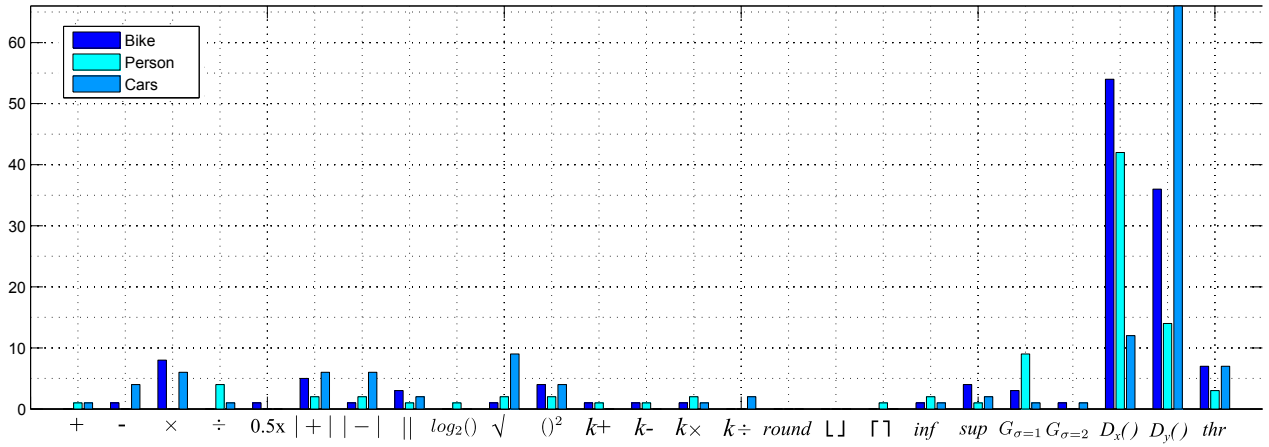


(a)

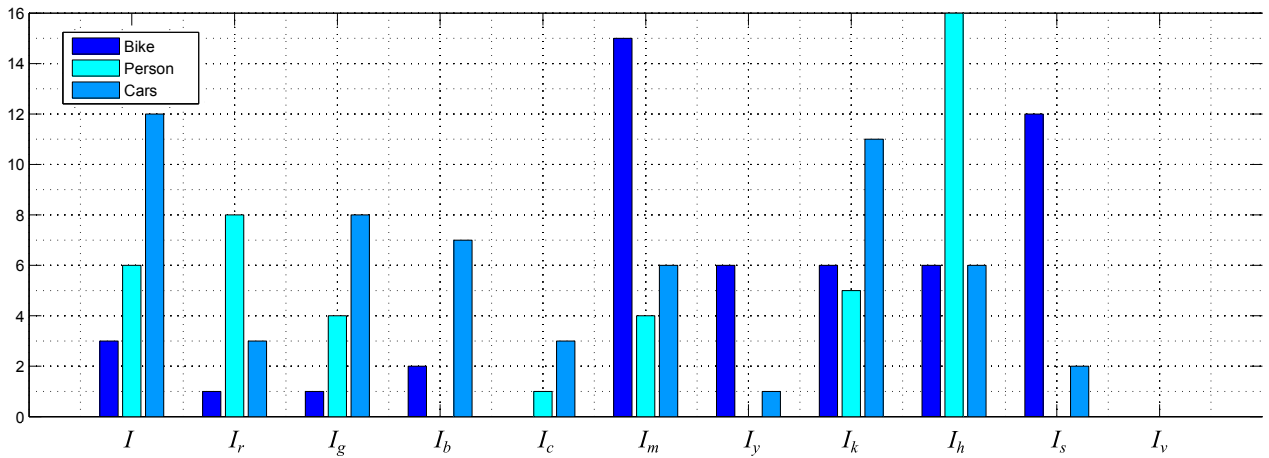


(b)

Figura 49: Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de color, utilizadas para crear las soluciones para las clases de GRAZ-02.



(a)



(b)

Figura 50: Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de orientación, utilizadas para crear las soluciones para las clases de GRAZ-02.

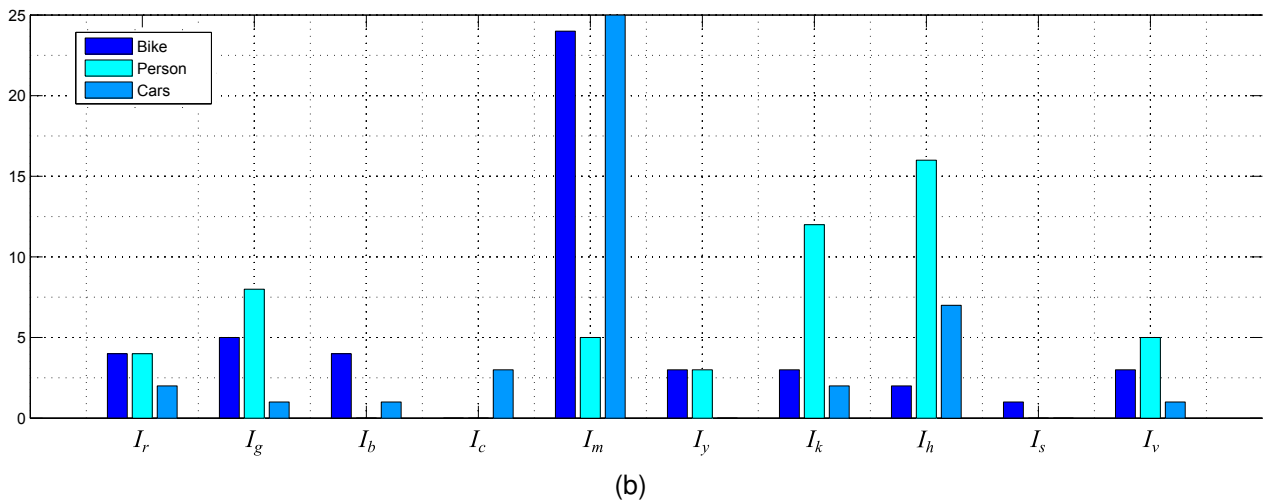
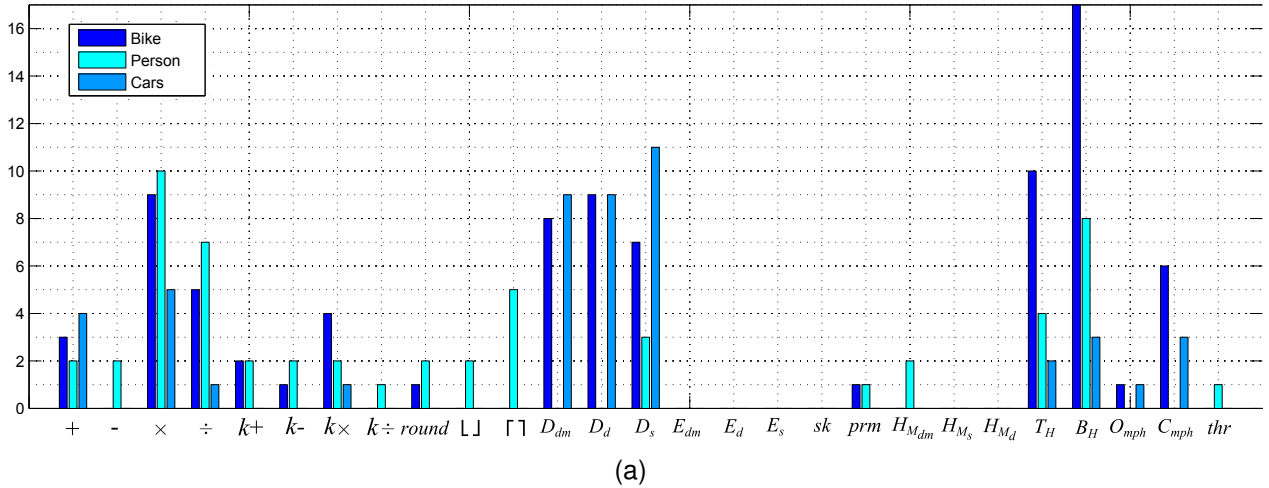


Figura 51: Gráficas de la frecuencia de uso de las funciones y terminales de la dimensión de forma, utilizadas para crear las soluciones para las clases de GRAZ-02.

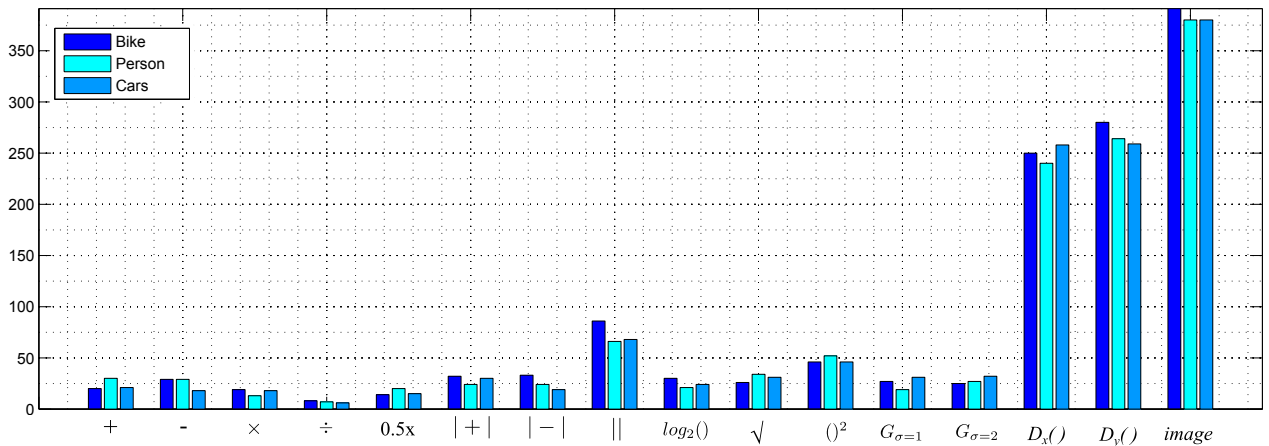
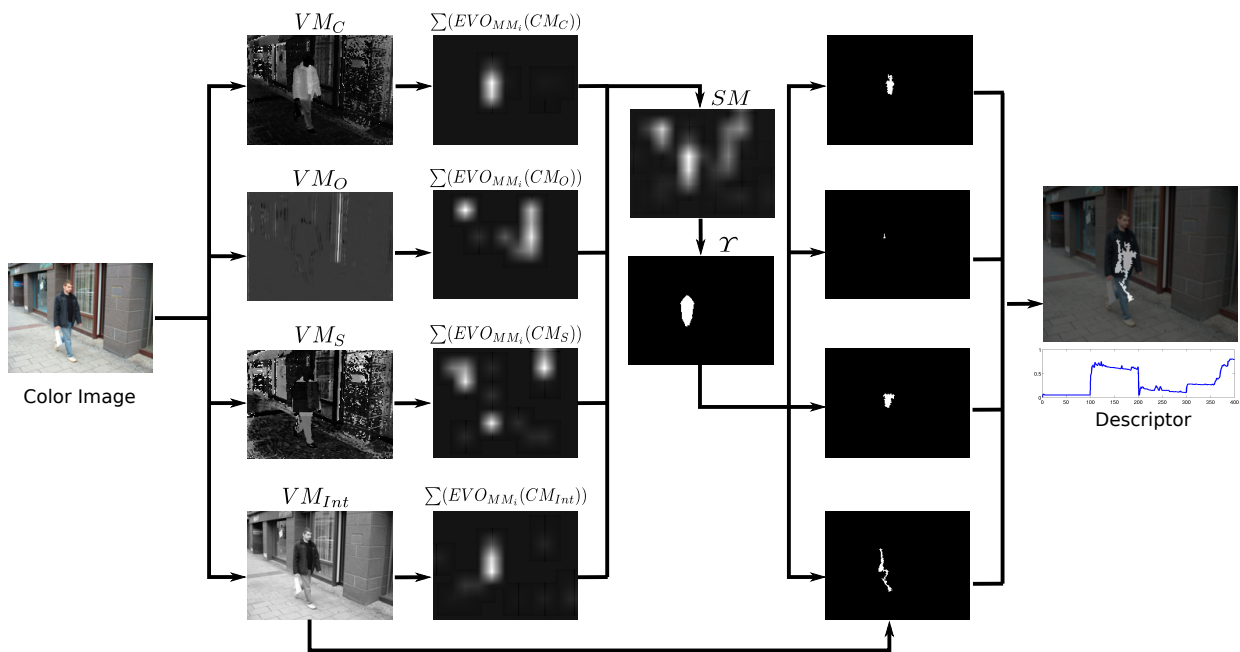


Figura 52: Gráficas de la frecuencia de uso de las funciones y terminales para los mapas mentales, utilizadas para crear las soluciones para las clases de GRAZ-02.



**Figura 53: Ejemplo del procesamiento de una solución.** Las imágenes representan el mapa de salida en cada una de las etapas del AVCMO. Esta solución en particular fue optimizada para clasificar la clase Persona de la base de datos GRAZ-02.

más detallado de los operadores que constituyen cada solución.

La Figura 53 presenta los mapas resultantes en cada etapa de la estructura jerárquica del AVCMO, con el objetivo de mostrar un ejemplo del funcionamiento de una solución optimizadas. Estos resultados corresponden al individuo señalado como *d*) en la Tabla 10. En la figura se muestran las transformaciones que sufre la imagen de entrada dentro de un individuo optimizado para la clasificación de la clase Persona. Como se observa en las imágenes, este individuo en particular solo utiliza información que pertenece al objeto de interés para la construcción del descriptor. En términos generales, esta solución logra buenos resultados de clasificación con un  $EER = 90.67\%$  de clasificación y un  $Average F\text{-measure} = 18\%$ . Así, esta solución supera a los sistemas del estado del arte. Es importante notar que la BP permite encontrar diferentes soluciones que logran buenos resultados, al explotar los beneficios de la evolución artificial, encontrando individuos dentro de un espacio de búsqueda extenso.

**Tabla 10: Estructura de los operadores correspondientes a los mejores soluciones para cada una de las clases de la base de datos GRAZ.**

Class	$EVO_d$ and $MM_k$ operators	Evaluation	
a)G01Bike	$OV_O = D_y(I_y) \times D_{yy}(I_y)$ $OV_C = (I_h)^2$ $OV_F = I_y \oplus SE_s$ $OV_{MM_1} = \log(D_x(D_{xy}(MC_d)))$ $OV_{MM_2} = G_{\sigma=1}(D_{xy}(MC_d))$ $OV_{MM_3} =  D_x(MC_d) - D_y(MC_d) $ $OV_{MM_4} = MC_d$ $OV_{MM_5} = \log(D_{xy}(MC_d))$ $OV_{MM_6} = D_{xy}(MC_d) - D_x(MC_d)$ $OV_{MM_7} = \frac{D_{xx}(MC_d)}{2}$ $OV_{MM_8} = (D_{yy}(MC_d))^2$	Train - 81.0 Val. - 72.0 Test - 84.0	
b)G01Person	$OV_O = G_{\sigma=1}(D_x(I_r)) + 0.10$ $OV_C = I_h \times (Op_{b-y}(I) - thr(I_k))$ $OV_F = T_{hat}(T_{hat}([I_r]))$ $OV_{MM_1} = D_x(MC_d)$ $OV_{MM_2} = \frac{MC_d}{2}$ $OV_{MM_3} = D_{xx}(MC_d)$ $OV_{MM_4} =  MC_d + D_{xy}(MC_d) $ $OV_{MM_5} = \frac{G_{\sigma=1}(MC_d)}{2}$ $OV_{MM_6} = D_{xy}(MC_d)$ $OV_{MM_7} =  \frac{MC_d}{2} - ( D_{xy}(MC_d) - D_x(MC_d) ) $ $OV_{MM_8} = D_y(D_{xy}(MC_d))$	Train - 78.0 Val. - 66.0 Test - 76.0	
c)G02Bike	$OV_O = D_{xy}(I_m)$ $OV_C = I_h \times 0.45$ $OV_F = ((I_m \oplus SE_d) \times 0.95) \times (B_{hat}(T_{hat}(I_r)))$ $OV_{MM_4} = D_x(MC_d)$ $OV_{MM_1} = MC_d$ $OV_{MM_2} = \frac{D_{xx}(MC_d)}{2}$ $OV_{MM_3} =  G_{\sigma=2}(MC_d) - D_y(MC_d) $ $OV_{MM_5} =  D_y(MC_d) + ( D_{yy}(MC_d) - D_{xx}(MC_d) ) $ $OV_{MM_6} = D_y(D_{yy}(MC_d) - D_{xx}(MC_d))$ $OV_{MM_7} = (D_y(D_{xx}(MC_d)))^2$	Train - 89.33 Val. - 82.67 Test - 86.67	
d)G02Person	$OV_O = D_{xx}(I_h)$ $OV_C = \frac{(I_h \times I_h) \times I_h}{I_r} / 0.14$ $OV_F = (I_g \oplus SE_d) \oplus SE_d$ $OV_{MM_1} = \frac{\log(D_{xy}(MC_d))}{D_y(MC_d) / D_y(MC_d)}$ $OV_{MM_2} = \log(\sqrt{D_{xy}(MC_d)})$ $OV_{MM_3} = MC_d$ $OV_{MM_4} = (D_{yy}(MC_d))^2$ $OV_{MM_5} =  MC_d $ $OV_{MM_6} =  D_{yy}(MC_d) $ $OV_{MM_7} = D_{xy}(MC_d)$ $OV_{MM_8} = D_{yy}(MC_d) - D_y(MC_d)$ $OV_{MM_9} = G_{\sigma=1}(MC_d)$	Train - 87.33 Val. - 89.33 Test - 90.67	
e)G02Cars	$OV_O = \sqrt{D_{yy}(I_b)}$ $OV_C = I_h$ $OV_F = I_m \oplus SE_d$ $OV_{MM_1} = D_y(\log(D_{xy}(MC_d) \times  MC_d - D_{xx}(MC_d) ))$ $OV_{MM_2} =  MC_d + MC_d $ $OV_{MM_3} = MC_d$ $OV_{MM_4} = G_{\sigma=2}( D_x(MC_d) - D_y(MC_d) )$ $OV_{MM_5} = \frac{MC_d}{2}$ $OV_{MM_6} = MC_d$ $OV_{MM_7} = D_y(D_{xy}(MC_d))$	Train. - 83.33 Val. - 81.33 Test - 85.33	



## Capítulo 6. Conclusiones y trabajo futuro

---

En este trabajo de investigación se propuso un sistema inspirado en el proceso que se lleva a cabo en la corteza visual humana, como un clasificador de objetos, llamado AVCMO. El modelo propuesto sigue la estructura jerárquica descrita propuesta por diversos modelos computacionales, emulando el flujo de información de las rutas dorsal y ventral de sistema natural a través de un esquema funcional. Reemplazando el uso de porciones de imágenes para describir las clases de objetos con un conjunto de operadores visuales. El AVCMO se basa en los modelos psicológicos, fisiológicos, y principalmente computacionales de los trabajos descritos en (Fukushima, 1987; Olshausen *et al.*, 1993; Walther *et al.*, 2002; Itti *et al.*, 1998; Riesenhuber y Poggio, 1999; Serre *et al.*, 2005; Mutch y Lowe, 2008; Wersing y Körner, 2003; Ghodrati *et al.*, 2012; Huang *et al.*, 2011).

Además, se tomo el modelo de la atención visual como estrategia para localizar el objeto de interés, es decir, definir la región de la imagen que ocupa. En base a esta región, se selección los puntos más prominentes en la imagen con el objetivo de crear un vector descriptor. A diferencia de los modelos previos, el AVCMO utiliza la información visual contenida en los mapas visuales (primera etapa del sistema), dado que la información ha sufrido menos transformaciones comparado con los mapas mentales. De esta manera, se integra las tareas de detección y descripción de objetos dentro de un mismo modelo. Es decir, la descripción se hace en base a una región de la imagen, en vez de toda la superficie de la misma.

Se extendió el algoritmo evolutivo conocido como programación cerebral a un ambiente multi-objetivo; usando dos criterios de evaluación: la razón de error equivalente para medir el rendimiento en la tarea de clasificación, y la medida-F para revisar la calidad de las soluciones para localizar el objeto de interés. Por medio de la integración del sistema SPEA2 para los proceso de selección y supervivencia. Después de analizar los resultados del proceso de validación cruzada, se puede concluir que esta metodología es viable para el desarrollo de sistemas de clasificación. El esquema multi-objetivo permite encontrar mejores resultados en el espacio de soluciones.

Siguiendo el protocolo de entrenamiento y pruebas propuesto en el trabajo de Opelt *et al.* (2006), usando 150 imágenes de entrenamiento, 75 de validación y 75 de prueba;

con el objetivo de comparar el sistema propuesto con otros del estado-del-arte. La evaluación del rendimiento de clasificación de los sistemas se hizo por medio de la razón de error equivalente, EER, utilizada como métrica estándar en las bases de datos GRAZ-01 y GRAZ-02. Los resultados obtenidos muestran que el modelo propuesto logra resultados similares a los del estado-del-arte; superándolos en algunas instancias. El rendimiento del AVCMO se comparó con sistemas de clasificación tradicionales como los descritos en el trabajo de Opelt *et al.* (2006), así como con diversos sistemas bio-inspirados (Mutch y Lowe, 2008; Huang *et al.*, 2011; Ghodrati *et al.*, 2012).

### 6.1. Aportaciones de la tesis

- La contribución principal de esta tesis es la integración de las tareas de detección y descripción de un objeto para clasificarlo. Esto se logra al abordar el problema de clasificación de objetos como un problema de optimización con un enfoque multi-objetivo. Dicho enfoque se fundamenta en la combinación de los enfoques mono-objetivo desarrollados en los trabajos de Dozal *et al.* (2014) y Olague *et al.* (2014a).
- Se desarrolló una nueva estrategia para la creación del vector descriptor de los objetos. Usando el paradigma del foco de atención para encontrar la región de la imagen donde se encuentra el objeto de interés, para posteriormente extraer la información visual de las primeras etapas del AVCMO donde la información del objeto es más abundante.
- El modelo de la AVCMO fue implementado para clasificar las conjuntos Persona y Bicicleta de la base de imágenes GRAZ-01; así como las clases Persona, Bicicleta y Carros que conforman la base de imágenes GRAZ-02. Estas bases de datos forman parte de los retos de clases de objetos visuales de la red Europea PASCAL, lo que permitió comparar los resultados obtenidos con modelos del estado-del-arte, tales como Opelt *et al.* (2006); Mutch y Lowe (2008); Huang *et al.* (2011); Ghodrati *et al.* (2012). Cabe notar, que de las cinco clases estudiadas, el modelo de la AVCMO logra resultados similares al estado-del-arte en las dos clases del set GRAZ-01; mientras que este supera al resto de los modelos estudiados en las tres clases de GRAZ-02.

- Se presentó un estudio de validación cruzada con el objetivo de evitar resultados engañosos, que podrían ser consecuencia del fenómeno conocido como sobre entrenamiento. Así, se muestra empíricamente la viabilidad del sistema de programación cerebral multi-objetivo como un esquema de entrenamiento para un clasificador.

## 6.2. Trabajo futuro

De esta manera, partiendo de los resultados obtenidos a lo largo de este trabajo de investigación, se pueden realizar diferentes proyectos futuros:

- Una de los primeros pasos podría ser, realizar un estudio exhaustivo de los modelos creados por medio de la evolución artificial. Realizando un estudio formal de cada uno de los operadores que componen una o varias soluciones. Y de esta manera medir la estabilidad de los operadores a los diferentes tipos de transformaciones, lineales, afines o proyectivas; además de otros efectos visuales producto de diversos tipos de ruido.
- En segundo plano, extender la experimentación a bases de imágenes más complejas, como es el caso del reto VOC (Everingham *et al.*, 2010) o el reto de reconocimiento de objetos a gran escala de ImageNet (Russakovsky *et al.*, 2015).
- Modificar el modelo para conciderar múltiples vistas en la tarea de descripción de los objetos, y así poder abordar problemas de clasificación multi-vista, como los presentes en la base de imágenes de NORB (LeCun *et al.*, 2004).
- En base a la relación tan estrecha que tienen las tareas de localización y descripción, resulta interesante analizar el modelo desde una perspectiva coevolutiva, presentando así un modelo que siga el paradigma de las dos rutas corticales del sistema natural.
- Implementar el modelo en un ambiente paralelo como CUDA, con el objetivo de reducir el tiempo de procesamiento del sistema al tratar imágenes grandes.

## Lista de referencias bibliográficas

- Al-Sahaf, H., Zhang, M., y Johnston, M. (2015). Binary image classification: A genetic programming approach to the problem of limited training instances. *Evolutionary Computation*, (20): 1–40.
- Behnamian, J. (2014). A parallel competitive colonial algorithm for {JIT} flowshop scheduling. *Journal of Computational Science*, 5(5): 777 – 783.
- Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94: 115–147.
- Bleuler, S., Laumanns, M., Thiele, L., y Zitzler, E. (2003). Pisa — a platform and programming language independent interface for search algorithms. En: C. Fonseca, P. Fleming, E. Zitzler, L. Thiele, y K. Deb (eds.), *Evolutionary Multi-Criterion Optimization*, Vol. 2632 de *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 494–508.
- Chan, S. (2001). Complex adaptive systems. En: *ESD.83 Research Seminar in Engineering Systems*. pp. 1–9.
- Clancey, W. J. (1984). Classification problem solving. Reporte técnico, Department of Computer Science, Stanford University. STAN-CS-84-1018.
- Clemente, E. (2015). *Reconocimiento de objetos en una escena bajo el paradigma del cómputo evolutivo y la corteza visual*. Tesis de doctorado, Centro de Investigación Científica y de Educación Superior de Ensenada, Ensenada, Baja California. 199 pp.
- Clemente, E., Olague, G., Dozal, L., y Mancilla, M. (2012). Object recognition with an optimized ventral stream model using genetic programming. En: C. Di Chio, A. Agapitos, S. Cagnoni, C. Cotta, F. de Vega, G. Di Caro, R. Drechsler, A. Ekárt, A. Esparcia-Alcázar, M. Farooq, W. Langdon, J. Merelo-Guervós, M. Preuss, H. Richter, S. Silva, A. Simões, G. Squillero, E. Tarantino, A. Tettamanzi, J. Togelius, N. Urquhart, A. Uyar, y G. Yannakakis (eds.), *Applications of Evolutionary Computation*, Vol. 7248 de *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 315–325.
- Clemente, E., Olague, G., Hernández, D., L. Briseño, J., y Mercado, J. (2015). Object detection in natural images using the brain programming paradigm with a multi-objective approach. En: A. M. Mora y G. Squillero (eds.), *Applications of Evolutionary Computation*, Vol. 9028 de *Lecture Notes in Computer Science*. Springer International Publishing, pp. 201–213.
- Corne, D. W., Knowles, J. D., y Oates, M. J. (2000). The pareto envelope-based selection algorithm for multiobjective optimization. En: *Parallel Problem Solving from Nature PPSN VI*. Springer, pp. 839–848.
- Darwin, C. y Bynum, W. F. (1859). *On the origin of species by means of natural selection: or, the preservation of favored races in the struggle for life*. John Murray, London. p. 502.
- Deb, K., Agrawal, S., Pratap, A., y Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. *Lecture notes in computer science*, 1917: 849–858.

- Deng, J., Ding, N., Jia, Y., Frome, A., Murphy, K., Bengio, S., Li, Y., Neven, H., y Adam, H. (2014). Large-scale object classification using label relation graphs. En: *Computer Vision–ECCV 2014*. Springer, pp. 48–64.
- Deriche, R. (1993). Recursively implementing the gaussian and its derivatives. Reporte técnico, Institut National de Recherche en Informatique et Automatique. TR-1893.
- Deriche, R. y Giraudon, G. (1993). A computational approach for corner and vertex detection. *International Journal of Computer Vision.*, **2**(10): 101–124.
- Desimone, R. y Duncan, J. (1995a). Neural mechanisms of selective visual attention. *Annual review of neuroscience*, **18**(1): 193–222.
- Desimone, R. y Duncan, J. (1995b). Neural mechanisms of selective visual attention. *Annual Review of Neuroscience*, **18**(1): 193–222. PMID: 7605061.
- Dozal, L., Olague, G., Clemente, E., y Sánchez, M. (2012). Evolving visual attention programs through evo features. En: *Applications of Evolutionary Computation*. Springer, pp. 326–335.
- Dozal, L., Olague, G., Clemente, E., y Hernández, D. (2014). Brain programming for the evolution of an artificial dorsal stream. *Cognitive Computation*, **6**(3): 528–557.
- Dunn Rivera, E. (2006). *Técnicas evolutivas multiobjetivo para el posicionamiento de cámaras de un sistema de metrología basado en visión*. Tesis de doctorado, Centro de Investigación Científica y de Educación Superior de Ensenada, Ensenada, Baja California. 209 pp.
- Eiben, A. y Smith, J. E. (2003). *Introduction to Evolutionary Computing*. Springer. p. 300.
- Eigen, M. (1973). *Evolutionsstrategie Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. mit einem Nachwort von Manfred Eigen, Friedrich Frommann Verlag, Struttgart-Bad Cannstatt. p. 170.
- Everingham, M., Van Gool, L., Williams, C. K., Winn, J., y Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, **88**(2): 303–338.
- Farivar, R. (2009). Dorsal–ventral integration in object recognition. *Brain Research Reviews*, **61**(2): 144 – 153.
- Fei-Fei, L., Fergus, R., y Perona, P. (2007). Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, **106**(1): 59–70.
- Fogel, L. J., Owens, A. J., y Walsh, M. J. (1966). *Artificial Intelligence Through Simulated Evolution*. John Wiley and Sons. p. 170.
- Fukushima, K. (1987). Neural network model for selective attention in visual pattern recognition and associative recall. *Applied Optics*, **23**(26): 4985–92.

- Ghodrati, M., Khaligh-Razavi, S.-M., Ebrahimpour, R., Rajaei, K., y Pooyan, M. (2012). How can selection of biologically inspired features improve the performance of a robust object recognition model? *PLoS ONE*, **7**(2): e32357.
- Hernández, D. E., Clemente, E., Olague, G., y Briseño, J. L. (2015). Evolutionary multi-objective visual cortex for object classification in natural images. *Journal of Computational Science*. Publicación pendiente.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan Press. Michigan, USA, p. 183.
- Holland, J. (1992). Complex adaptive systems. *A New Era in Computation*, **1**(121): 17–30.
- Holland, J. (2006). Studying complex adaptive systems. *Journal of Systems Science and Complexity*, **19**(1): 1–8.
- Horaud, R. y Monga, O. (1995). *Vision par ordinateur - outils fondamentaux*. Hermès, segunda edición. p. 425.
- Huang, Y., Huang, K., Tao, D., Tan, T., y Li, X. (2011). Enhanced biologically inspired model for object recognition. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, **41**(6): 1668–1680.
- Hubel, D. H. (1982). Exploration of the primary visual cortex, 1955-78. *Nature*, **5883**(299): 515–524.
- Hubel, D. H. y Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology*, **148**(3): 574–591.
- Hubel, D. H. y Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, **160**(1): 106–154.
- Itti, L. (2003). Modeling primate visual attention. *Computational Neuroscience: A Comprehensive Approach.*, **2**(10): 635–655.
- Itti, L. y Koch, C. (2001). Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging.*, **10**(1): 161–169.
- Itti, L., Koch, C., y Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **20**(11): 1254–1259.
- Klette, R. (2014). *Concise Computer Vision - An Introduction into Theory and Algorithms*. Undergraduate Topics in Computer Science. Springer. London, UK, p. 429.
- Koch, C. y Ullman, S. (1985). Shifts in selective visual attention: Towards the underlying neural circuitry. *Human Neurobiology.*, **2**(4): 219–227.
- Koza, J. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. The MIT Press. Boston, USA, p. 819.

- LeCun, Y., Huang, F. J., y Bottou, L. (2004). Learning methods for generic object recognition with invariance to pose and lighting. En: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on. IEEE*, Vol. 2, pp. 11–19.
- Lin, C.-H. y Lin, P.-L. (2014a). Improving the non-dominated sorting genetic algorithm using a gene-therapy method for multi-objective optimization. *Journal of Computational Science*, **5**(2): 170 – 183.
- Lin, C.-H. y Lin, P.-L. (2014b). Improving the non-dominated sorting genetic algorithm using a gene-therapy method for multi-objective optimization. *Journal of Computational Science*, **5**(2): 170–183.
- Logothetis, N. K. y Sheinberg, D. L. (1996). Visual object recognition. *Annual review of neuroscience*, **19**(1): 577–621.
- MacLeod, D. I. A. y Boynton, R. M. (1979). Chromaticity diagram showing cone excitation by stimuli of equal luminance. *J. Opt. Soc. Am.*, **69**(8): 1183–1186.
- Marr, D. (1982). *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc. New York, NY, USA, p. 432.
- Marr, D. y Hildreth, E. (1980). Theory of edge detection. *Proceedings of the Royal Society of London B: Biological Sciences*, **207**(1167): 187–217.
- Milner, A. y Goodale, M. (2006). *The visual brain in action*. Oxford University Press, segunda edición. p. 320.
- Mishkin, M., Ungerleider, L., y Macko, K. (1983). Object vision and spatial vision: two cortical pathways. *Trends in Neurosciences*, **6**: 414–417.
- Mutch, J. y Lowe, D. (2008). Object class recognition and localization using sparse features with limited receptive fields. *International Journal of Computer Vision*, **80**(1): 45–57.
- Olague, G. (2016). *Evolutionary Computer Vision: The First Footprints*. Natural Computing Series, Springer. p. 420.
- Olague, G. y Trujillo, L. (2011). Evolutionary-computer-assisted design of image operators that detect interest points using genetic programming. *Image Vision Comput.*, **29**(7): 484–498.
- Olague, G. y Trujillo, L. (2012). Interest point detection through multiobjective genetic programming. *Applied Soft Computing*, **12**(8): 2566 – 2582.
- Olague, G., Clemente, E., Dozal, L., y Hernández, D. (2014a). Evolving an artificial visual cortex for object recognition with brain programming. En: O. Schuetze, C. A. Coello Coello, A.-A. Tantar, E. Tantar, P. Bouvry, P. D. Moral, y P. Legrand (eds.), *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation III*, Vol. 500 de *Studies in Computational Intelligence*. Springer International Publishing, pp. 97–119.

- Olague, G., Dozal, L., Clemente, E., y Ocampo, A. (2014b). Optimizing an artificial dorsal stream on purpose for visual attention. En: O. Schuetze, C. A. Coello Coello, A.-A. Tantar, E. Tantar, P. Bouvry, P. D. Moral, y P. Legrand (eds.), *EVOLVE - A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation III*, Vol. 500 de *Studies in Computational Intelligence*. Springer International Publishing, pp. 141–166.
- Olshausen, B., Anderson, C., y Essen, D. V. (1993). A neurobiological model of visual attention and invariant pattern recognition based on dynamic routing of information. *The Journal of Neuroscience*, **11**(13): 4700–19.
- Opelt, A., Pinz, A., Fussenegger, M., y Auer, P. (2006). Generic object recognition with boosting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **28**(3): 416–431.
- Otsu, N. (1975). A threshold selection method from gray-level histograms. *Automatica*, **11**(285-296): 23–27.
- Pajares M, G. y De La Cruz Garcia, J. M. (2008). *Visión por computador. Imágenes Digitales y Aplicaciones*. p. 764.
- Pareto, V. (1897). The new theories of economics. *The Journal of Political Economy*, **5**(4): 485–502.
- Perez, C. B. y Olague, G. (2009). Evolutionary learning of local descriptor operators for object recognition. En: *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*. ACM, pp. 1051–1058.
- Pinto, N., Cox, D. D., y DiCarlo, J. J. (2008). Why is real-world visual object recognition hard? *PLoS Comput Biol*, **4**(1): e27.
- Poli, R., Langdon, W. B., McPhee, N. F., y Koza, J. R. (2008). *A field guide to genetic programming*. Lulu. com.
- Ranftl, R. y Pock, T. (2014). A deep variational model for image segmentation. En: X. Jiang, J. Hornegger, y R. Koch (eds.), *Pattern Recognition*, Vol. 8753 de *Lecture Notes in Computer Science*. Springer International Publishing, pp. 107–118.
- Reid, P. (2004). *Biometrics for network security*. Prentice Hall Professional. p. 252.
- Riesenhuber, M. y Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature neuroscience*, **2**(11): 1019–1025.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., y Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, **115**(3): 211–252.
- Schulz, H. y Behnke, S. (2012). Learning object-class segmentation with convolutional neural networks. En: *11th European Symposium on Artificial Neural Networks (ESANN)*. Vol. 3, pp. 1–6.



- Serre, T., Kouh, C., Cadieu, M., Knoblich, G., Kreiman, U., y Poggio, T. (2005). Theory of object recognition: computations and circuits in the feedforward path of the ventral stream in primate visual cortex. Reporte técnico, MIT-CSAIL. MIT-CSAIL-TR-2005-082.
- Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., y Poggio, T. (2007). Robust object recognition with cortex-like mechanisms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **29**(3): 411–426.
- Shao, L., Liu, L., y Li, X. (2014). Feature learning for image classification via multiobjective genetic programming. *Neural Networks and Learning Systems, IEEE Transactions on*, **25**(7): 1359–1371.
- Silva, S. y Almeida, J. (2003). Gplab-a genetic programming toolbox for matlab. En: *Proceedings of the Nordic MATLAB Conference*. pp. 273–278.
- Treisman, A. y Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, **1**(12): 97–136.
- Trujillo, L. (2008). *Cómputo evolutivo aplicado en el diseño de métodos para la detección y descripción de regiones de interés*. Tesis de doctorado, Centro de Investigación Científica y de Educación Superior de Ensenada, Ensenada, Baja California. 271 pp.
- Trujillo, L. y Olague, G. (2008). Automated design of image operators that detect interest points. *Evol. Comput.*, **16**: 483–507.
- Ungerleider, L. G. (1982). Two cortical visual systems. *Analysis of visual behavior*, pp. 549–586.
- Von Neumann, J., Burks, A. W., *et al.* (1966). Theory of self-reproducing automata. *IEEE Transactions on Neural Networks*, **5**(1): 3–14.
- Wagner, M., Cai, W., Lees, M., y Aydt, H. (2015). Evolving agent-based models using self-adaptive complexification. *Journal of Computational Science*, **2015**(10): 351–359.
- Walther, D. y Koch, C. (2006). 2006 special issue: Modeling attention to salient proto-objects. *Neural Netw.*, **19**(9): 1395–1407.
- Walther, D., Itti, L., Riesenhuber, M., Poggio, T., y Koch, C. (2002). Attentional selection for object recognition – a gentle way. En: *Proceedings of the Second International Workshop on Biologically Motivated Computer Vision*. Springer-Verlag, BMCV '02, pp. 472–479.
- Wersing, H. y Körner, E. (2003). Learning optimized features for hierarchical models of invariant object recognition. *Neural Comput*, **7**(15): 1559–1588.
- Zitzler, E., Laumanns, M., Thiele, L., Zitzler, E., Zitzler, E., Thiele, L., y Thiele, L. (2001). Spea2: Improving the strength pareto evolutionary algorithm. Reporte técnico.