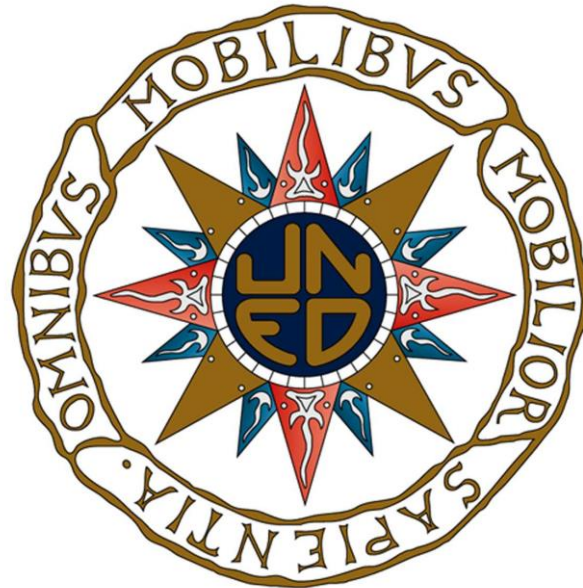


**UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA**  
Escuela Técnica Superior de Ingeniería Informática  
Departamento de Inteligencia Artificial

---



---

**USO DEL ALGORITMO *DIFFERENTIAL EVOLUTION* Y  
EL DESCRIPTOR *LOCAL BINARY PATTERNS* PARA EL  
RECONOCIMIENTO DE OBJETOS EN IMÁGENES  
DIGITALES**

---

TRABAJO FIN DE MÁSTER EN INTELIGENCIA ARTIFICIAL AVANZADA:  
SISTEMAS INTELIGENTES DE DIAGNÓSTICO, PLANIFICACIÓN Y CONTROL

Miguel Revuelta Espinosa

Director: Dr. Enrique Javier Carmona Suárez

Septiembre 2017

## Agradecimientos

No sabía ni siquiera sabía si me iba a dar tiempo a poder escribir todos los agradecimientos que merecen tantas personas por unas razones u otras.

En primer lugar quiero mostrarles mi gratitud a mis padres, Bernardo y María Jesús; que desgraciadamente he tenido que visitarlos a ambos al hospital durante el transcurso de esta investigación. Gracias por vuestro apoyo incondicional, sin él dudo que hubiese podido seguir adelante.

Continúo agradeciendo su apoyo a mis hermanos Guillermo y Jorge, y mi cuñada Esther, que han sido especialmente cuidadosos con dejarme espacio y siempre han mostrado interés por saber si me estaba yendo bien. Gracias a vosotros también.

Sofía, mi sobrinita, merece una mención especial. Lamento profundamente no haber podido asistir a tu primer cumpleaños, no por ti que no vas a acordarte de ello, sino porque es un momento que me he perdido.

A mis amigos, Miguel; gracias por entender que no disponía del tiempo que nos habría gustado dedicar a tomarnos unas cervezas y charlar sobre Bakunin o Kant, Anyuri; gracias por haber estado siempre cerca para animarme a continuar y también gracias por comprender que no me podía apuntar a esos planes de ocio tan interesantes que organizas de vez en cuando, Olga; sin ti y la serenidad que transmites dudo mucho que no hubiera perdido los papeles en alguna ocasión. También gracias por dejar que me concentrase en esta actividad y haberme guiado espiritualmente en todo momento.

Mis demás amistades, gracias también por estar en mi vida, vuestro apoyo y ser tolerantes con la escasa disponibilidad que os he ofrecido durante este proyecto.

No me olvido de mis antiguos profesores, gracias por haberme ayudado a formarme y gracias también por haberme dejado abiertas las puertas a otros posibles proyectos para el futuro.

Enrique, mi tutor, tú también mereces una mención en estos breves agradecimientos. Además de haberme aceptado en el Máster, hecho esencial para poder cursarlo, has vivido conmigo muchos momentos de crisis y desesperación por ver que en algunas de las ideas que probábamos no obteníamos los resultados previstos. Gracias también por no haberte rendido en ningún momento y la paciencia que has mostrado hacia mi persona cuando no me enteraba bien de algún concepto. Creo que eres la única persona que se hace una verdadera idea de todo el trabajo y dedicación que se esconden detrás de este texto, además del tuyo quiero decir. Es un lástima que esta memoria solo sea un reflejo de todo ese esfuerzo.

# CONTENIDO

---

Agradecimientos .....	II
Ecuaciones .....	V
Figuras .....	VI
Ilustraciones .....	VII
Tablas.....	VIII
Capítulo 1. Introducción.....	2
1.1. Motivación .....	3
1.2. Descripción del Problema .....	3
1.3. Limitación o alcance del estudio .....	4
1.4. Hipótesis y Objetivos de Investigación .....	5
1.5. Estructura del resto de la memoria.....	6
Capítulo 2. Estado del Arte .....	8
2.1. Introducción .....	9
2.2. Operadores de Visión.....	10
2.2.1. Operadores de detección de esquinas .....	10
2.2.2. Operadores para la detección de bordes .....	13
2.2.3. Operadores de detección de regiones o blobs.....	15
2.2.4. Operador LBP.....	20
2.3. Metaheurísticas y Operadores de Visión .....	25
Capítulo 3. Descripción de la metodología .....	34
3.1. Conocimiento Previo .....	35
3.1.1. Descripción Operador LBP .....	35
3.1.2. Differential Evolution.....	41
3.2. Descripción del método .....	49
3.2.1. Codificación de los individuos .....	50
3.2.2. Proceso de decodificación de los individuos.....	50
3.2.3. Función Fitness.....	51
Capítulo 4. Resultados Experimentales y Discusión.....	52
4.1. Consideraciones previas .....	53

4.1.1.	Estadísticos propios del Algoritmo Evolutivo.....	53
4.1.2.	Descripción de los campos de las tablas de resultados.....	54
4.2.	Experimento 1: Exploración del caso base .....	55
4.2.1.	Objetivo .....	55
4.2.2.	Configuración del experimento .....	55
4.2.3.	Análisis de los resultados .....	57
4.3.	Experimento 2: Exploración de diferentes tipos de templates.....	60
4.3.1.	Objetivo .....	60
4.3.2.	Configuración del Experimento.....	60
4.3.3.	Análisis de los resultados .....	63
4.4.	Experimento 3: Influencia de la variación de intensidad luminosa .....	67
4.4.1.	Objetivo .....	68
4.4.2.	Configuración del Experimento.....	68
4.4.3.	Análisis de los resultados .....	71
4.5.	Experimento 4: Influencia del ruido gaussiano .....	75
4.5.1.	Objetivo .....	75
4.5.2.	Configuración del Experimento.....	75
4.5.3.	Análisis de los resultados .....	79
4.6.	Experimento 5: Influencia del ruido sal y pimienta.....	80
4.6.1.	Objetivo .....	81
4.6.2.	Configuración del Experimento.....	81
4.6.3.	Análisis de los resultados .....	84
4.7.	Experimento 6: Influencia del color y decodificación para imágenes RGB....	87
4.7.1.	Objetivo .....	87
4.7.2.	Configuración del Experimento.....	88
4.7.3.	Análisis de los resultados .....	92
4.8.	Experimento 7: Combinación de diferentes Operadores LBP .....	94
4.8.1.	Objetivo .....	95
4.8.2.	Configuración del experimento .....	95
4.8.3.	Análisis de los resultados .....	98

4.9.	Experimento 8: Selección de un subconjunto de bins del histograma LBP ..	100
4.9.1.	Objetivo .....	100
4.9.2.	Configuración del experimento .....	101
4.9.3.	Análisis de los resultados .....	104
4.10.	Experimento 9: Búsqueda Local .....	106
4.10.1.	Objetivo .....	106
4.10.2.	Configuración del Experimento .....	107
4.10.3.	Análisis de los resultados .....	107
Capítulo 5.	Uso del método en un problema real .....	112
5.1.	Consideraciones previas .....	114
5.2.	Objetivo .....	115
5.3.	Configuración del Experimento .....	115
5.3.1.	Template .....	115
5.3.2.	Umbral de la Tasa de Éxito .....	116
5.4.	Análisis de los resultados .....	116
5.4.1.	Tasa de éxito promediada .....	116
5.4.2.	Distancias intercentrales .....	118
5.4.3.	MBF .....	118
5.4.4.	Duración de las ejecuciones y los runs .....	119
5.4.5.	Mejores Fitness .....	119
5.4.6.	Resultados sobre las imágenes de retina .....	119
Capítulo 6.	Conclusiones y trabajo futuro .....	124
6.1.	Conclusiones .....	125
6.2.	Trabajo Futuro .....	126
Referencias	.....	128
Anexos	.....	134
Anéxo A:	Tablas de Resultados .....	136
A.1:	Resultados Decodificación Básica .....	136
A.2:	Resultados Exploración templates homogéneos .....	136
A.3:	Resultados Influencia de la intensidad luminosa .....	137

A.4:	Resultados Influencia del ruido gaussiano .....	138
A.5:	Resultados Influencia del ruido sal y pimienta .....	140
A.6:	Resultados Influencia del color y decodificación para imágenes RGB .....	141
A.7:	Resultados Combinación de diferentes Operadores LBP .....	141
A.8:	Resultados Selección de un subconjunto de bins del histograma LBP .....	142
A.9:	Resultados Búsquedas Locales.....	142
A.10:	Resultados Disco Óptico .....	144
Anexo B:	Procesos de decodificación detallados.....	146
B.1:	Consideraciones previas .....	146
B.2:	Decodificación para el problema básico .....	147
B.3:	Decodificación para el problema de template escalado .....	149
B.4:	Decodificación para el problema con rotación.....	152
B.5:	Decodificación para el problema del template escalado y rotado.....	155
Anexo C:	Visión Artificial y Computación Evolutiva.....	159
	Visión Artificial.....	159
	Computación Evolutiva.....	162

## Ecuaciones

Ecuación 3.1.1-1:	Formulación del patrón LBP de un píxel .....	38
Ecuación 3.1.1-2:	Binarización para la diferencia de intensidades de un vecindario ....	38
Ecuación 3.1.1-3:	Construcción del vector o descriptor LBP de una imagen .....	38
Ecuación 3.1.2-1:	Donnor Vector.....	44
Ecuación 3.1.2-2:	Trial Vector .....	45
Ecuación 3.1.2-3:	Survival Vector .....	45
Ecuación 3.2.3-1:	Función Fitness básica .....	51
Ecuación 3.2.3-2:	Resto (píxeles que sobran) .....	51
Ecuación 4.1.1-1:	Umbral para el criterio de éxito.....	53
Ecuación 4.1.1-2:	Tasa de Éxito (SR) .....	53
Ecuación 4.1.1-3:	Mean Best Fitness .....	54
Ecuación 4.7.1-1:	Función Fitness para imágenes a color .....	87
Ecuación 3.2.3-4:	Función Fitness promedio de dos LBP .....	94
Ecuación 4.8.3-5:	Función Fitness que aplica un reajuste al descriptor LBP .....	100

Ecuación 4.8.3-2: Función de reajuste para descriptor LBP .....	100
Ecuación 5.3.2-1: Radio del Disco Óptico .....	116

## Figuras

Figura 3.1.1-1: Vecindario de un píxel y sus niveles de intensidad luminosa .....	36
Figura 3.1.1-2: Diferencia de intensidades de un píxel central respecto a su vecindario	36
Figura 3.1.1-3: Vecindario binarizado.....	36
Figura 3.1.1-4: Proceso del Operador LBP .....	37
Figura 3.1.1-5: Coordenadas de una matriz imagen .....	37
Figura 3.1.1-6: Descriptores LBP según los parámetros P y R .....	39
Figura 3.1.1-7: Ejemplo de descriptor LBP uniforme .....	40
Figura 3.1.1-8: Rotaciones de un mismo patrón LBP .....	41
Figura 3.1.2-1: Proceso Differential Evolution .....	45
Figura 3.1.2-2: Ejemplo de población inicial en DE .....	47
Figura 3.1.2-3: Obtención del Weighed Differential Vector .....	48
Figura 3.1.2-4: Obtención del Donnor Vector.....	48
Figura 3.1.2-5: Recombinación en DE .....	49
Figura 3.1.2-6: Superviviente para la siguiente generación .....	49
Figura 3.2.2-1: Esquema a alto nivel del método propuesto .....	50
Figura 4.3.3-1: Comparación template pilas 250GR vs 350GR experimento 2.....	66
Figura 4.3.3-2: Comparación template rueda 250GR vs 350GR experimento 2 .....	67
Figura 4.5.3-1: Pérdida de rendimiento Ruido Gaussiano .....	80
Figura 4.6.3-1: Pérdida de rendimiento sal y pimienta.....	86
Figura 4.7.3-1: Comparación SR decodificación RGB .....	93
Figura 4.7.3-2: Comparación del promedio SR gris y color .....	94
Figura 4.8.3-1: Comparación SR promediado con función que combina diferentes LBP .....	99
Figura 4.9.3-1: Comparación SR promediado de la función fitness original respecto a la que considera un subconjunto de bins del histograma .....	105
Figura 4.10.3-1: Comparación SR promediado estrategias de búsqueda local .....	110
Figura 5.4.1-1: Histograma DO tasa de éxito.....	117
Figura 5.4.1-2: Diagrama de tartas DO .....	117
Figura 5.4.2-1: Distancias intercentrales DO .....	118
Figura 5.4.3-1: DO MBF y std (MBF) .....	118
Figura 5.4.5-1: DO mejores fitness encontrados .....	119
Figura B.1-1: Matriz imagen en Matlab .....	146
Figura B.2-1: Problema básico, provisión de escena y template.....	147

Figura B.2-2: Problema básico, significado de los cromosomas x e y .....	148
Figura B.2-3: Problema básico, dimensiones del individuo imagen .....	148
Figura B.2-4: Extracción del individuo imagen y cálculo de su LBP .....	149
Figura B.2-5: Problema básico, asignación de fitness .....	149
Figura B.3-1: Problema con cambio de escala, provisión de escena y template .....	150
Figura B.3-2: Problema con cambio de escala, significado de los cromosomas x e y .	151
Figura B.3-3: Problema con cambio de escala, dimensiones del individuo imagen ....	151
Figura B.3-4: Problema con cambio de escala, extracción del individuo imagen y cálculo de su LBP .....	152
Figura B.4-1: Problema con rotación, provisión de escena y template .....	153
Figura B.4-2: Problema con rotación, significado de los cromosomas x e y .....	153
Figura B.4-3: Problema con rotación, dimensiones del individuo imagen .....	154
Figura B.4-4: Problema con rotación, extracción 1 .....	154
Figura B.4-5: Problema con rotación, extracción 2 .....	155
Figura B.5-1: Problema con escalado y rotación, provisión de escena y template .....	156
Figura B.5-2: Problema con escalado y rotación, significado de los cromosomas x e y .....	156
Figura B.5-3: Problema con rotación y escalado, dimensiones de extracción .....	157
Figura B.5-4: Problema con escalado y rotación, significado del cromosoma $\theta$ .....	157
Figura B.5-5: Problema con escalado y rotación, significado del cromosoma $\alpha$ .....	158

## Ilustraciones

Ilustración 2.2.1-1: Puntos de interés obtenidos por el Algoritmo FAST .....	11
Ilustración 2.2.1-2: Puntos de interés obtenidos por el filtro Harris.....	12
Ilustración 2.2.1-3: Puntos de interés obtenidos por el Algoritmo BRISK .....	13
Ilustración 2.2.2-1: Detector de Sobel .....	14
Ilustración 2.2.2-2: Ejemplo detector de Canny .....	14
Ilustración 2.2.3-1: Regiones clasificadas por el Algoritmo MSER .....	15
Ilustración 2.2.3-2: Características obtenidas por el descriptor HOG .....	16
Ilustración 2.2.3-3: Características detectadas por el Algoritmo SURF.....	18
Ilustración 2.2.3-4: Características encontradas por el algoritmo por el Algoritmo FREAK .....	19
Ilustración 2.2.3-5: Descriptor LBP y su aplicación a una imagen .....	20
Ilustración 4.2.2-1: Escena experimento exploración del caso base .....	56
Ilustración 4.3.2-1: Escena experimento 2 .....	61
Ilustración 4.3.2-2: Template circuito e histograma LBPri en experimento 2 .....	62
Ilustración 4.3.2-3: Template pilas e histograma LBPri del experimento 2.....	62



Ilustración 4.3.2-4: Template rueda e histograma LBPri del experimento 2 .....	63
Ilustración 4.3.3-1: Óptimo local hallado en experimento 2 .....	65
Ilustración 4.4.2-1: Escena con variación de intensidad uniforme $\gamma=2.0$ .....	69
Ilustración 4.4.2-2: Escena con una variación de intensidad uniforme $\gamma=1.0$ .....	70
Ilustración 4.4.2-3: Escena con variación de intensidad uniforme $\gamma=0.5$ .....	70
Ilustración 4.5.2-1: Escena experimento 4, $\gamma=0.1$ .....	76
Ilustración 4.5.2-2: Escena experimento 4, $\gamma=0.01$ .....	77
Ilustración 4.5.2-3: Escena experimento 4, $\gamma=0.001$ .....	78
Ilustración 4.6.2-1: Escena con ruido sal y pimienta $\gamma=0.01$ .....	82
Ilustración 4.6.2-2: Escena con ruido sal y pimienta $\gamma=0.05$ .....	83
Ilustración 4.6.2-3: Escena con ruido sal y pimienta $\gamma=0.1$ .....	83
Ilustración 4.7.2-1: Escena a color .....	88
Ilustración 4.7.2-2: Vista de los canales RGB del template circuito y el histograma LBP de cada canal.....	89
Ilustración 4.7.2-3: Vista de los canales RGB del template pilas y el histograma LBP de cada canal .....	90
Ilustración 4.7.2-4: Vista de los canales RGB del template rueda y el histograma LBP de cada canal .....	90
Ilustración 4.8.2-1: Escena experimento 7 .....	96
Ilustración 4.8.2-2: Vista template circuito con diferentes histogramas LBP .....	97
Ilustración 4.8.2-3: Vista template pilas con diferentes histogramas LBP.....	97
Ilustración 4.8.2-4: Vista template rueda con diferentes histogramas LBP .....	98
Ilustración 4.9.2-1: Escena experimento 8 .....	101
Ilustración 4.9.2-2: Vista template circuito con histograma LBP ajustado .....	102
Ilustración 4.9.2-3: Vista template pilas con histograma LBP ajustado.....	103
Ilustración 4.9.2-4: Vista template rueda con histograma LBP ajustado .....	103
Ilustración 5.4.6-1: DIARETDB1 resultado en imagen de retina, índice 51.....	120
Ilustración 5.4.6-2: DIARETDB1 resultado en imagen de retina, índice 80.....	121
Ilustración 5.4.6-3: DIARETDB1 resultado en imagen de retina, índice 84.....	122

## Tablas

Tabla 3.1.1-1: Patrón LBP rotado.....	40
Tabla 3.2.1-1: Modelos de genotipo .....	50
Tabla 4.2.2-1: Templates experimento 1 .....	57
Tabla 4.2.3-1: Resumen de las mejores combinaciones F y CR .....	59
Tabla 4.4.2-1: Templates experimento 3 .....	71
Tabla 4.5.2-1: Templates experimento 4 .....	78

Tabla A.1-1: Resultados decodificación básica TLF .....	136
Tabla A.1-2: Resultados decodificación básica TLFS .....	136
Tabla A.1-3: Resultados decodificación básica TLFR.....	136
Tabla A.1-4: Resultados decodificación básica TLFSR.....	136
Tabla A.2-1: Resultados exploración templates homogéneos, pilas GR=250 .....	136
Tabla A.2-2: Resultados exploración templates homogéneos, rueda GR=250 .....	136
Tabla A.2-3: Resultados exploración templates homogéneos, pilas GR=350 .....	137
Tabla A.2-4: Resultados exploración templates homogéneos, rueda GR=350 .....	137
Tabla A.3-1: Resultados influencia de la intensidad luminosa, circuito $\gamma=2.0$ .....	137
Tabla A.3-2: Resultados influencia de la intensidad luminosa, pilas $\gamma=2.0$ .....	137
Tabla A.3-3: Resultados influencia de la intensidad luminosa, rueda $\gamma=2.0$ .....	137
Tabla A.3-4: Resultados influencia de la intensidad luminosa, circuito $\gamma=1.0$ .....	137
Tabla A.3-5: Resultados influencia de la intensidad luminosa, pilas $\gamma=1.0$ .....	138
Tabla A.3-6: Resultados influencia de la intensidad luminosa, rueda $\gamma=1.0$ .....	138
Tabla A.3-7: Resultados influencia de la intensidad luminosa, circuito $\gamma=0.5$ .....	138
Tabla A.3-8: Resultados influencia de la intensidad luminosa, pilas $\gamma=0.5$ .....	138
Tabla A.3-9: Resultados influencia de la intensidad luminosa, rueda $\gamma=0.5$ .....	138
Tabla A.4-1: Resultados influencia ruido gaussiano, circuito $\gamma=0.001$ .....	138
Tabla A.4-2: Resultados influencia ruido gaussiano, pilas $\gamma=0.001$ .....	138
Tabla A.4-3: Resultados influencia ruido gaussiano, rueda $\gamma=0.001$ .....	139
Tabla A.4-4: Resultados influencia ruido gaussiano, circuito $\gamma=0.01$ .....	139
Tabla A.4-5: Resultados influencia ruido gaussiano, pilas $\gamma=0.01$ .....	139
Tabla A.4-6: Resultados influencia ruido gaussiano, rueda $\gamma=0.01$ .....	139
Tabla A.4-7: Resultados influencia ruido gaussiano, circuito $\gamma=0.1$ .....	139
Tabla A.4-8: Resultados influencia ruido gaussiano, pilas $\gamma=0.1$ .....	139
Tabla A.4-9: Resultados influencia ruido gaussiano, rueda $\gamma=0.1$ .....	139
Tabla A.5-1: Resultados influencia ruido sal y pimienta, circuito $\gamma=0.01$ .....	140
Tabla A.5-2: Resultados influencia ruido sal y pimienta, pilas $\gamma=0.01$ .....	140
Tabla A.5-3: Resultados influencia ruido sal y pimienta, rueda $\gamma=0.01$ .....	140
Tabla A.5-4: Resultados influencia ruido sal y pimienta, circuito $\gamma=0.05$ .....	140
Tabla A.5-5: Resultados influencia ruido sal y pimienta, pilas $\gamma=0.05$ .....	140
Tabla A.5-6: Resultados influencia ruido sal y pimienta, rueda $\gamma=0.05$ .....	140
Tabla A.5-7: Resultados influencia ruido sal y pimienta, circuito $\gamma=0.1$ .....	141
Tabla A.5-8: Resultados influencia ruido sal y pimienta, pilas $\gamma=0.1$ .....	141
Tabla A.5-9: Resultados influencia ruido sal y pimienta, rueda $\gamma=0.1$ .....	141
Tabla A.6-1: Resultados influencia del color, circuito.....	141
Tabla A.6-2: Resultados influencia del color, pilas.....	141
Tabla A.6-3: Resultados influencia del color, rueda .....	141
Tabla A.7-1: Resultados combinación de diferentes Operadores LBP, circuito.....	142

Tabla A.7-2: Resultados combinación de diferentes Operadores LBP, pilas.....	142
Tabla A.7-3: Resultados combinación de diferentes Operadores LBP, rueda .....	142
Tabla A.8-1: Resultados selección de un subconjunto de bins del histograma LBP, circuito.....	142
Tabla A.8-2: Resultados selección de un subconjunto de bins del histograma LBP, pilas .....	142
Tabla A.8-3: Resultados selección de un subconjunto de bins del histograma LBP, rueda .....	142
Tabla A.9-1: Resultados búsqueda local fixed single, circuito .....	143
Tabla A.9-2: Resultados búsqueda local fixed single, pilas .....	143
Tabla A.9-3: Resultados búsqueda local fixed single, rueda.....	143
Tabla A.9-4: Resultados búsqueda local threshold single, circuito.....	143
Tabla A.9-5: Resultados búsqueda local threshold single, pilas .....	143
Tabla A.9-6: Resultados búsqueda local threshold single, rueda .....	143
Tabla A.9-7: Resultados búsqueda local fixed multi, circuito .....	143
Tabla A.9-8: Resultados búsqueda local fixed multi, pilas .....	144
Tabla A.9-9: Resultados búsqueda local fixed multi, rueda.....	144
Tabla A.10-1: Resultados Retina Disco Óptico.....	145

# **CAPÍTULO 1. INTRODUCCIÓN**

---

## **1.1. Motivación**

A lo largo de las últimas décadas, el incremento del número de imágenes digitales disponibles ha crecido de forma exponencial gracias al perfeccionamiento de las redes de comunicaciones digitales y al auge de los teléfonos inteligentes, que disponen la mayor parte de ellos de cámaras digitales de última generación.

Todos los días se generan cientos de millones de imágenes nuevas, por lo que es preciso disponer de métodos que permitan analizar dichas imágenes en un tiempo razonable.

La capacidad de delegar la interpretación de las imágenes digitales a Sistemas Inteligentes es crucial para numerosos campos donde no existe el suficiente personal humano para dicha interpretación, como podría ser la medicina o la defensa.

Por ello es preciso desde el ámbito de la investigación científica continuar explorando vías y plantear propuestas innovadoras que supongan una mejora a las técnicas y metodologías contemporáneas características de la Visión Artificial.

## **1.2. Descripción del Problema**

La detección automática de objetos es la actividad de encontrar un objeto dado en imágenes digitales o secuencias de vídeo.

En Visión Artificial, realizar esta actividad supone un reto notable y retador: la aparición de artefactos o ruido en la imagen, cambios de orientación, tamaño, color, iluminación del objeto dado, la oclusión de parte del objeto a localizar en la escena, o la aparición de objetos similares son algunas de los motivos que convierten la detección de objetos en un problema complejo.

Incluso podría buscarse un objeto en una escena, a partir de un modelo similar a dicho objeto.

En el contexto de este estudio, se analizan algunos de estos problemas, donde a partir de un template que contiene un objeto, dicho objeto es buscado por una imagen escena añadiendo progresivamente algunas de las dificultades del problema mencionadas anteriormente.

En primer lugar, el formato más simple de problema de detección de un objeto en una escena, es aquel donde dicho objeto pertenece a la escena, sin que se aprecien transformaciones afines, fotométricas ni de ningún tipo.

En el siguiente tipo de problema, existe un cambio de escala entre el objeto buscado y objeto dado. De manera similar, otro problema es aquel en el que existe un cambio de orientación en el plano imagen entre el objeto template y ese mismo objeto en la escena.

Dentro de los casos más complejos abarcados en el estudio, se aborda el problema de localizar un objeto en una escena que puede concordar con un modelo de objeto al que se le ha aplicado tanto cambio de escala como rotación.

Por último se analiza un caso donde el modelo de objeto no pertenece a ninguna de las escenas donde se busca un objeto parecido al modelo.

Como tipo de problema en sí, cabe añadir que la variación de intensidad luminosa también quedaría contemplada de forma subyacente y complementaria a todos estos tipos de problema descritos anteriormente.

Hay que tener en cuenta una serie de consideraciones adicionales, que también influyen en la complejidad del problema.

La primera, es determinar los modelos de los objetos o templates: estos pueden ser representados como una mera matriz imagen, o bien aprovechar alguna de las representaciones producidas por la aplicación de operadores de Visión Artificial, es decir servirse de una representación compacta del modelo de objeto.

La elección del tipo de representación puede influir en la velocidad a la que los objetos son localizados así como en la precisión encontrando dichos objetos.

También es preciso determinar qué criterio de similitud entre un template y un hipotético objeto establecer para considerar si realmente ha sido localizado el objeto que deseaba encontrarse en la escena o no.

Este criterio influye también influye en la complejidad del problema y la precisión.

Por último, el procedimiento para localizar el objeto, es decir, cómo buscar el objeto en la escena tiene un impacto en el tiempo dedicado a encontrar el objeto. El método de búsqueda más simple y menos eficiente sería recorrer la escena de principio a fin.

### **1.3. Limitación o alcance del estudio**

Debido a la inmensidad de posibilidades de afrontar el problema planteado: cómo representar los objetos, cómo buscarlos en la imagen escena, qué criterio de similitud emplear para encontrar una solución válida, además de muchos otros factores ajenos al problema en sí:

El alcance del estudio se limita a emplear una única Metaheurística como motor de búsqueda del template en la escena, la cual es Differential Evolution, la aplicación de un único Operador de Visión para obtener una representación compacta del objeto a localizar, el operador LBP.

De forma adicional, se realiza un estudio del Estado del Arte que incluye breves descripciones de distintos tipos de Operadores de Visión, acompañados algunos con sumarios sobre estudios donde son aplicados: pero no entra dentro del alcance realizar una comparativa entre otros Operadores y el LBP. un breve resumen sobre la disciplina de las Metaheurísticas así como la descripción de algunas de sus variantes y estudios donde son aplicadas Metaheurísticas a distintos problemas de visión Artificial para el reconocimiento de objetos, especialmente si intervienen Operadores de Visión.

Sobre el Operador LBP se realiza una revisión más detenidamente, señalando y describiendo estudios recientes donde éste o alguna de sus variedades, tiene una relevancia especial.

Se describe el método empleado, y se realizan una serie de experimentos para validar la capacidad de la metodología propuesta dentro del contexto de cada uno de los problemas mencionados en la sección Descripción del Problema.

## **1.4. Hipótesis y Objetivos de Investigación**

La hipótesis principal de este estudio es comprobar la capacidad del Operador LBP como herramienta para localización de objetos en escenas. Este operador tradicionalmente ha sido empleado con el fin de detectar texturas, por lo tanto para objetos que tengan algún tipo de textura, la comparación de los descriptores LBP de dos imágenes que plasman un objeto similar, debería poder utilizarse como criterio de similitud entre dichas imágenes.

Para poder realizar una búsqueda de un objeto en una escena de forma eficiente ante distintos tipos de disposición del objeto en la escena y otros factores, se emplea Differential Evolution como motor de búsqueda.

La aplicación de una metodología que combina LBP con DE para resolver problemas de reconocimiento de objetos es el objetivo principal de la investigación.

De manera adyacente este objetivo, se analiza el rendimiento del operador LBP en diferentes situaciones, por ejemplo en escenas donde hay ruido.

También se analiza el comportamiento de la metodología en base a la parametrización del motor de búsqueda y la introducción de búsquedas locales, entre otros.

Para validar el objetivo principal de la investigación, la metodología presentada en este estudio es aplicada a un caso real, la búsqueda del disco óptico en una colección de imágenes de retina.

## **1.5. Estructura del resto de la memoria**

El resto de la memoria es estructurada del siguiente modo: en el capítulo segundo se ofrece un sumario del estado del Estado del Arte, donde se revisan algunos Operadores de Visión Artificial realizando una pausa especial con el Operador LBP y posteriormente se describen varios estudios relacionados con la aplicación de Metaheurísticas a problemas de Visión Artificial.

El tercer capítulo se destina primero a explicar detalladamente el Operador LBP y el algoritmo Differential Evolution y a continuación se explica la implementación o metodología diseñada para este estudio.

En el cuarto capítulo se presenta una colección de experimentos y los resultados obtenidos de los mismos con el fin de validar el comportamiento de la metodología ante distintos tipos de problemas de detección de objetos. En el quinto capítulo la metodología es aplicada a un caso real, la detección del disco óptico en imágenes de retina.

En el sexto capítulo se comparte una reflexión sobre las conclusiones el trabajo futuro.

Hay además una serie de anexos destinados a contribuir a la comprensión del estudio: en el primer anexo se adjuntan los resultados de todos los experimentos presentados y analizados en la memoria, el siguiente anexo incluye una descripción en detalle de la decodificación, el último anexo incluye un breve resumen sobre la disciplina de la Visión Artificial y la Computación Evolutiva.





## **CAPÍTULO 2. ESTADO DEL ARTE**

---

## 2.1. Introducción

En las últimas dos décadas, el aumento de la cantidad de imágenes digitales de todo tipo ha sido exponencial. Esto ha motivado numerosas vías de investigación dentro del ámbito de la Visión Artificial orientadas a poder sistematizar la interpretación de imágenes.

La sistematización de la interpretación de imágenes digitales es una actividad compleja cuyos objetivos casi siempre son dependientes del contexto del dominio de aplicación. No obstante, algunas de las fases de dicha sistematización, como pueden ser la detección de objetos u otras características de una imagen, tienen un carácter más generalizable.

El reconocimiento visual de objetos en imágenes digitales a través de la implementación de la implementación de diferentes técnicas, metodologías o procesos es un área de gran interés dentro del universo de la Visión Artificial.

Los principales objetivos del reconocimiento de objetos se enfocan en la mejora del rendimiento de los métodos existentes, la minimización del coste computacional de la aplicación de las distintas técnicas y mejorar la eficiencia en la fase de búsqueda de objetos en la imagen.

Los métodos de detección de objetos existentes suelen basarse en la aplicación de operadores de visión, algoritmos destinados a detectar y/o extraer características de una imagen, en numerosas ocasiones modelizando dichas características en una representación más compacta—por ejemplo, un vector en lugar de una matriz como sería una imagen—para poder así determinar con más rapidez y eficacia si un objeto aparece o no en una imagen dada.

Buscar uno, o peor aún varios objetos, en una imagen recorriéndola de principio a fin—es decir, por fuerza bruta—es una aproximación poco eficiente y dependiente del tamaño de la imagen, o peor aún; varias imágenes. Cuando se precisa detección en tiempo real, la fuerza bruta es impracticable. Por estas razones, la búsqueda de objetos en imágenes habitualmente se delega en estrategias de búsqueda propias de otras disciplinas.

En el campo de las Metaheurísticas pueden encontrarse numerosas familias de algoritmos que pueden ser utilizados y aprovechados como motores de búsqueda para diversos tipos de problemas diferentes, encontrando soluciones o candidatos a solución cercanos a los óptimos globales.

En la siguiente sección del capítulo se describen algunos Operadores de Visión que pueden emplearse para detectar características que faciliten la labor de detección de objetos y en la última sección se ofrece una revisión de distintas Metaheurísticas aplicadas en conjunto a distintos Operadores como propuesta para resolver problemas de detección de objetos.

## 2.2. Operadores de Visión

Aplicar filtros, detectores, u operadores es una tarea recurrente empleada cualquiera de los niveles para lograr el fin de encontrar características de una imagen que faciliten otros objetivos, como la detección de objetos o una interpretación de la propia imagen.

No existe una definición universal de lo que a una característica se refiere en los dominios de la Visión Artificial y el Procesamiento de Imágenes. Sin embargo, una noción podría ser la siguiente: “Una característica—en inglés, *feature*—es una ubicación de interés en la imagen”.

Estas ubicaciones de interés suelen consistir en puntos únicos como por ejemplo esquinas, bordes, regiones que comparten algún rasgo que las diferencia de las demás como por ejemplo su textura, o crestas, entre otros.

Mediante la aplicación de dichos operadores, es posible modelar descriptores: representaciones más compactas y eficientes de determinadas características de la imagen, para facilitar actividades como la detección de objetos, actividades, direcciones, escalas; en resumen, simplifican el análisis de la imagen.

### 2.2.1. Operadores de detección de esquinas

La detección de puntos de interés, *Feature Detection* es el proceso de adquisición de dichos puntos.

La extracción de características a menudo se realiza sobre unos puntos de interés localizados previamente, y consiste en estudiar las vecindades o regiones de los puntos de interés con el fin de generar representaciones más compactas, resistentes a transformaciones afines o fotométricas que describan la imagen en términos que permitan realizar con facilidad acciones tales como la localización de objetos o *Pattern Matching*.

Los algoritmos encomendados a estas tareas pertenecen a un conjunto de algoritmos conocidos como Operadores de Visión. A continuación, se describen brevemente algunos de ellos.

El primer ejemplo descrito es el detector de esquinas *Features from Accelerated Segment Test*, conocido como el Operador FAST, (Rosten & Drummond, 2005). Este método puede usarse para extraer puntos con el fin de mapear objetos en diversas tareas de Visión Artificial.

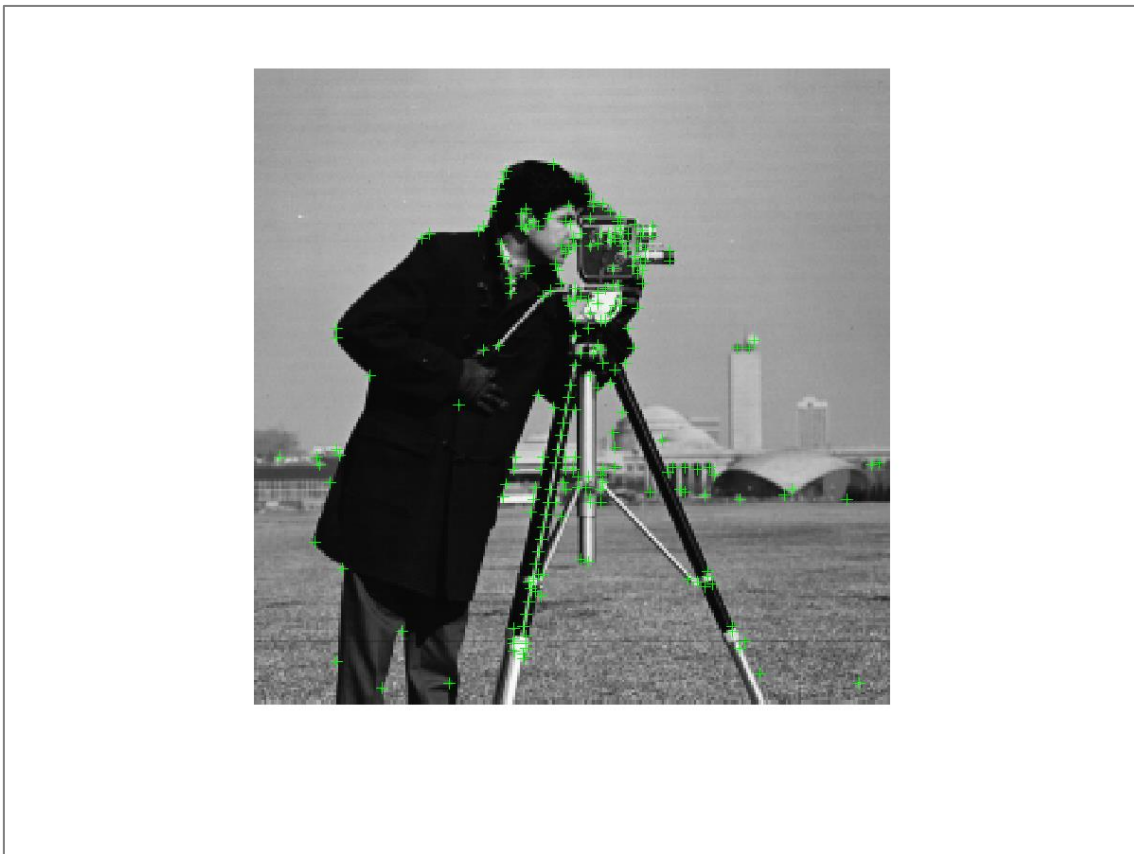
A partir de conjuntos de puntos pueden estudiarse propiedades geométricas de la imagen con más sencillez.

Una de las ventajas del método es su eficiencia computacional, siendo uno de los métodos conocidos más rápidos para la detección de esquinas.

Se basa en la utilización de vecindarios circulares de 16 píxeles para determinar si un punto candidato  $p$  es una esquina o no. Cada punto en el vecindario es etiquetado con un número entero en el sentido de las agujas del reloj en secuencia desde el 1 al 16.

Se considerará que el punto  $p$  es una esquina si en el conjunto de  $N$  píxeles contiguos a  $p$ , dentro del círculo del vecindario, todos los píxeles del conjunto  $N$  son o más brillantes o más tenues, en términos de intensidad luminosa que  $p$ .

La Ilustración 2.2.1-1 muestra una imagen sobre la que se han señalado todos los puntos de interés localizados en una escena tras aplicar el Operador FAST.



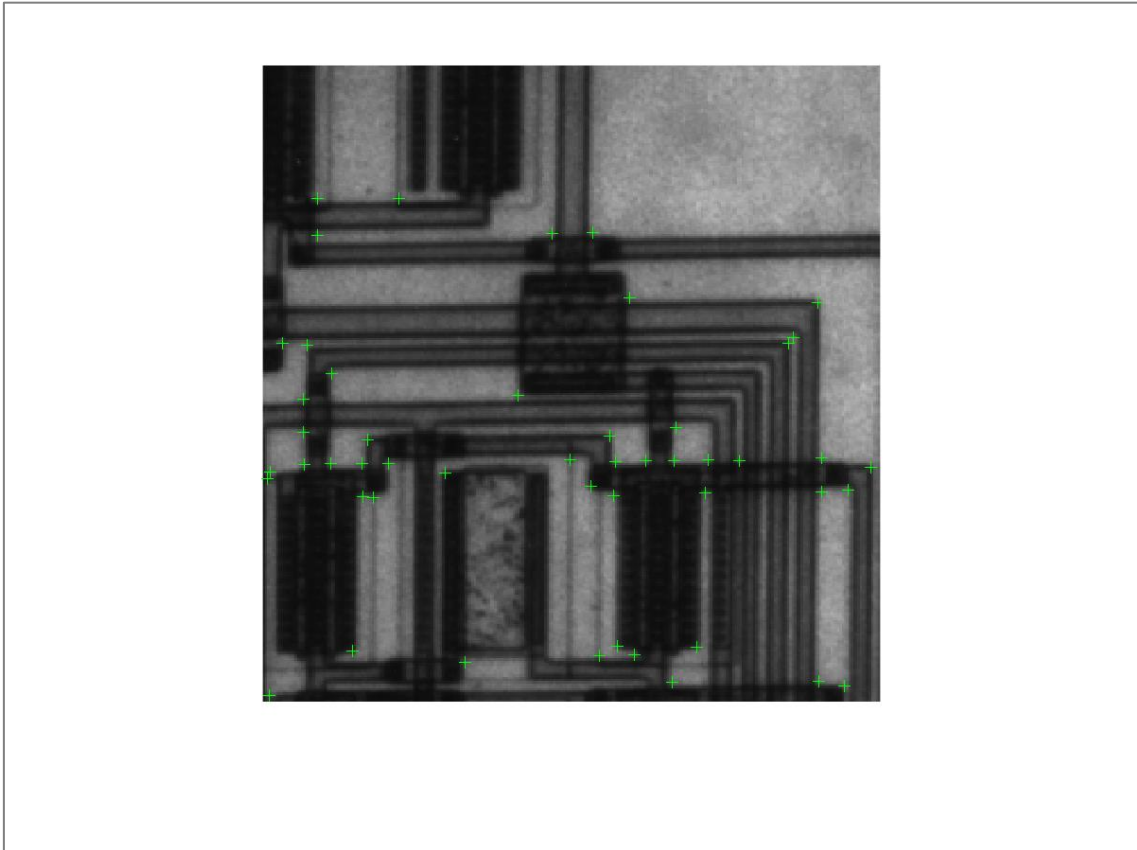
**Ilustración 2.2.1-1: Puntos de interés obtenidos por el Algoritmo FAST**

Otro popular detector de esquinas, es el de Harris (Muggleton-Harris & Brown, 1988), se trata de un operador de desplazamiento de ventanas que localiza características, concretamente esquinas, en una imagen. Es un método simple, rápido y popular por ser invariante a cambios de escala, iluminación y rotación.

En esencia, este operador identifica puntos de interés basándose en la introducción de una ventana sobre una imagen. Si dicha ventana es desplazada en cualquier dirección y existe una esquina dicha ventana, entonces se producirá un cambio brusco de intensidad.

Dado que se revisan todos los posibles desplazamientos de la ventana y se miden los cambios de intensidad producidos, es posible mediante este detector medir la “fuerza” o la certeza de que un punto sea una esquina, por lo que puede preestablecerse un criterio para solo etiquetar como esquinas aquellos píxeles que superen un determinado umbral.

En la Ilustración 2.2.1-2 ha sido aplicado el detector de Harris en una imagen de un circuito. Los puntos señalizados son un subconjunto de todos los encontrados de acuerdo a un umbral arbitrario.

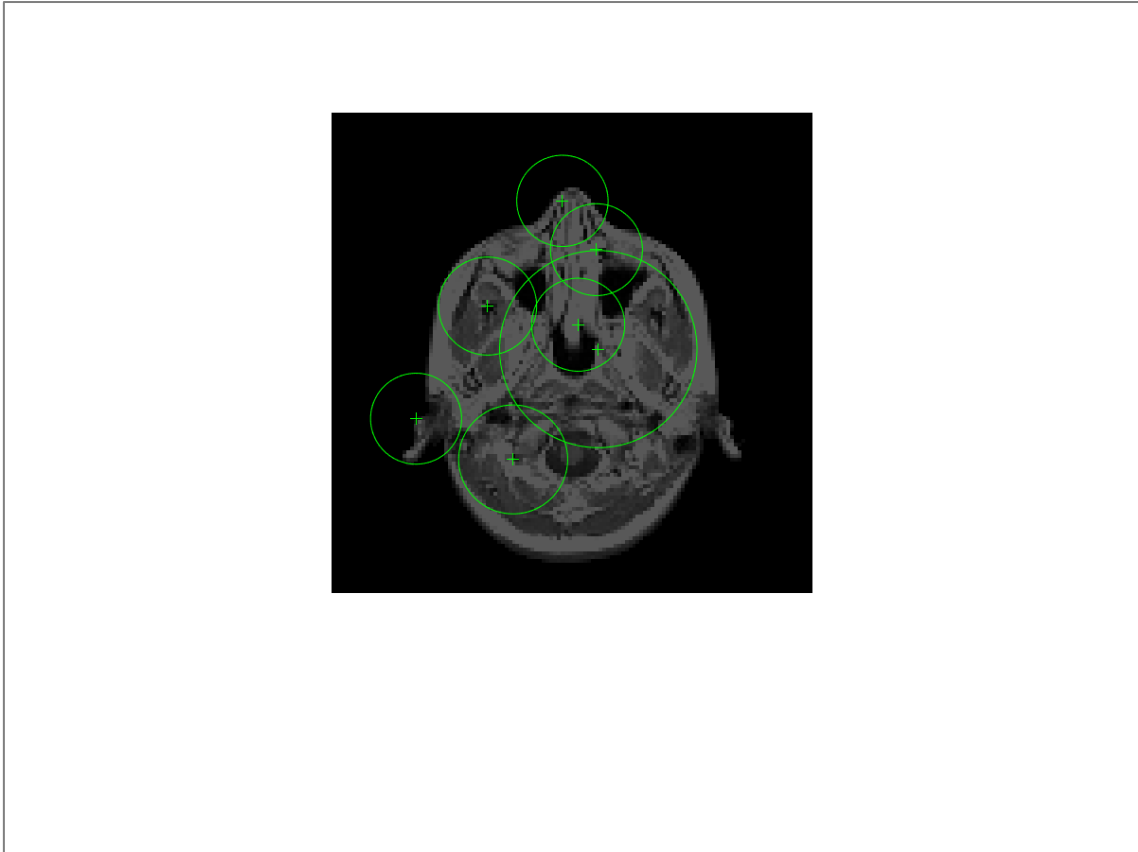


**Ilustración 2.2.1-2: Puntos de interés obtenidos por el filtro Harris**

Otro método, denominado *Binary Robust Invariant Scalable Keypoints* o sencillamente, BRISK (Leutenegger, Chli, & Siegwart, 2011) se basa en definir patrones circulares fáciles de configurar sobre la imagen, sobre los que computa comparaciones de intensidad luminosa en base a un descriptor binario.

El proceso comienza detectando características usando otro operador, como FAST o Harris, posteriormente realiza una composición de descriptores y por último una correlación de puntos clave.

En la Ilustración 2.2.1-3 se puede observar los puntos seleccionados por BRISK en una imagen de resonancia magnética, donde el radio de cada círculo manifiesta el grado de certeza según BRISK de que un punto sea una esquina; a mayor radio mayor certeza.



**Ilustración 2.2.1-3: Puntos de interés obtenidos por el Algoritmo BRISK**

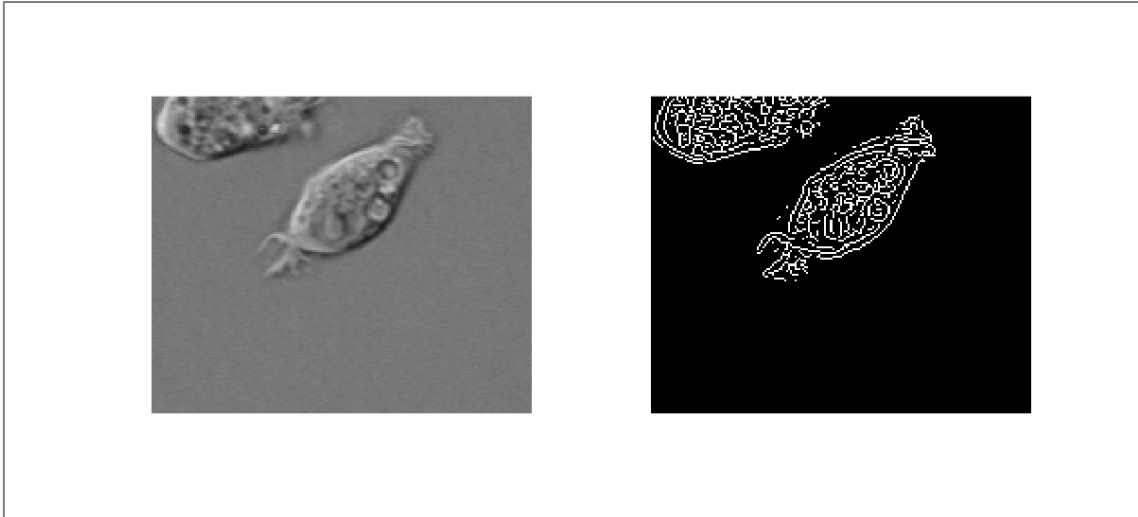
### **2.2.2. Operadores para la detección de bordes**

El detector de Sobel (Sobel, 1982) calcula el gradiente de intensidad de una imagen en cada píxel. Para ello es aplicada una máscara, una matriz 3x3, que recorre todos los píxeles de la imagen construyendo una imagen derivada.

Dada la configuración de la matriz máscara con la que se recorre la imagen original, los valores de los píxeles aparecen con intensidades que resaltan los bordes detectados en una imagen derivada, lo que permite realizar una binarización aproximada de los bordes respecto al resto de superficies.

En la Ilustración 2.2.2-1 puede apreciarse la potencia de este operador relativa a su capacidad para detectar bordes. La imagen de la izquierda es la imagen original de unas células, la de la derecha son los contornos que han sido producidos tras aplicar el operador.

La imagen derivada ha sido binarizada de modo que aquellos píxeles que el detector ha clasificado como “contorno” tienen el valor máximo de intensidad (que en binario es 1) y todos los demás el valor mínimo.



**Ilustración 2.2.2-1: Detector de Sobel**

El detector de Canny (Canny, 1986) presenta algunas similitudes respecto al detector de Sobel, pero se utiliza más máscaras y filtros de preprocesamiento.

En esencia, Canny emplea varios filtros, uno para eliminar ruido y otros cuatro filtros (máscaras 5x5) para detectar bordes en todas las direcciones.

En la Ilustración 2.2.2-2 puede observarse en la imagen de la derecha qué segmentos detectaría el operador sobre la imagen que aparece en la izquierda.

Cabe destacar la capacidad que exhibe operador a la hora de detectar bordes o segmentos que presentan curvatura.

En contrapartida, es posible identificar que los distintos brillos que aparecen en la imagen de la izquierda son “erróneamente” considerados bordes.



**Ilustración 2.2.2-2: Ejemplo detector de Canny**



### 2.2.3. Operadores de detección de regiones o blobs

Como ejemplo para la detección de blobs, destaca el algoritmo *Maximally Stable External Regions*, MSER (Matas, Chum, Urban, & Pajdla, 2004), empleado para encontrar correspondencias de elementos en dos imágenes obtenidas desde una perspectiva diferente.

El concepto subyacente en MSER reside en umbralizar sucesivamente cada imagen, obteniendo una secuencia de imágenes donde los píxeles cuya intensidad supera un umbral son convertidos a cero y los píxeles que son superiores a un segundo umbral son convertidos a 1—es decir, el umbral está definido por un intervalo—. Según se realizan las sucesivas umbralizaciones, las regiones de máxima intensidad y mínima intensidad van creciendo y fusionándose.

El conjunto final de todas las regiones conectadas es el conjunto de las regiones extremas. Dichas regiones extremas poseen dos propiedades útiles. La primera, que las coordenadas de transformación de las regiones son constantes, por lo que las regiones extremas son invariantes a transformaciones afines, no importa si la imagen es escalada, aplastada, o estrechada. La segunda propiedad es la transformación monótona de las intensidades de la imagen: el método es sensible a cambios de intensidad luminosa o proyecciones de sombras.

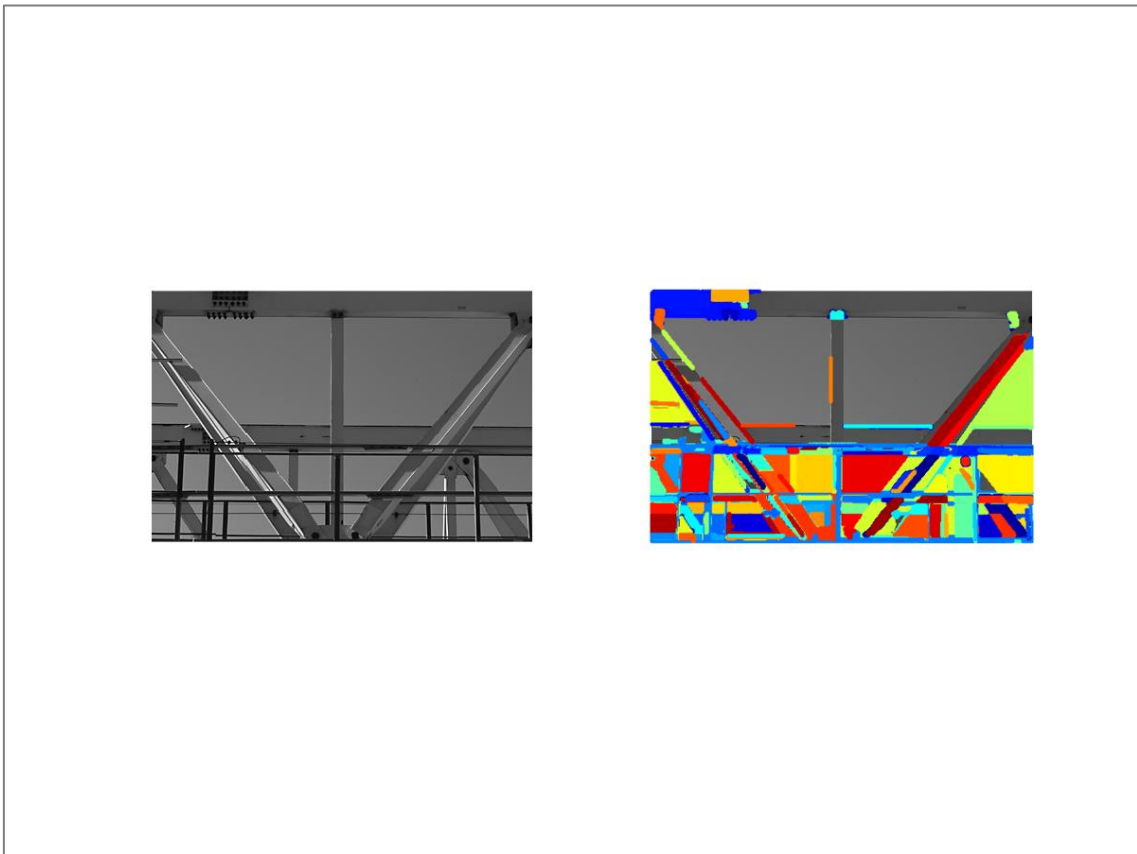
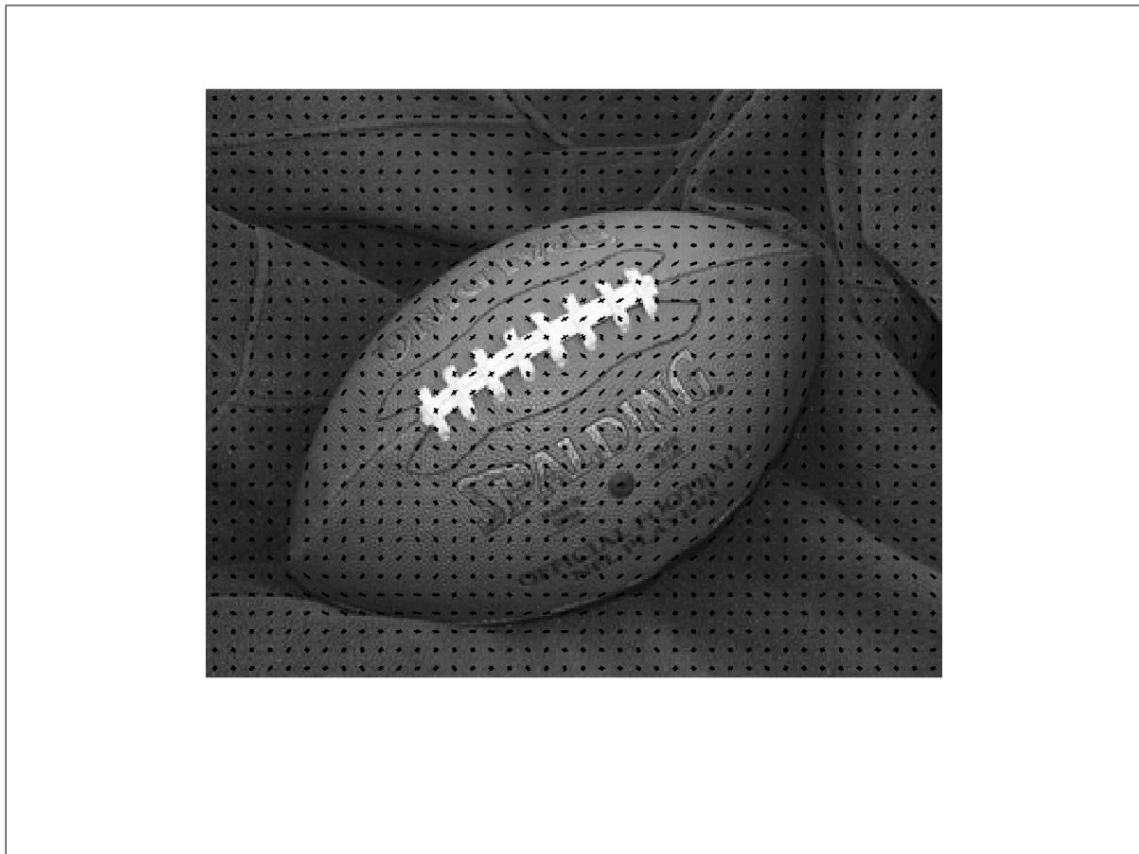


Ilustración 2.2.3-1: Regiones clasificadas por el Algoritmo MSER

El operador *Histogram of Oriented Gradients*, HOG (Dalai, 2005), es una técnica utilizada con el propósito de detectar objetos en imágenes digitales. Se basa en la contabilización de ocurrencias de orientaciones de gradiente localizados en regiones de la imagen.

La idea, es que la apariencia local de un objeto y su forma, dentro de una imagen, pueden ser descritas mediante la distribución de los gradientes o dirección de los bordes. Para ello, la imagen es dividida en celdas y para cada pixel de cada celda es calculado el histograma de gradientes. El descriptor resultante consiste en concatenar cada histograma.

Como ventajas del operador cabe destacar que es invariante a transformaciones geométricas (salvo rotación) y fotométricas. Suele ser empleado para detectar humanos en imágenes, especialmente peatones en imágenes de tráfico.



**Ilustración 2.2.3-2: Características obtenidas por el descriptor HOG**

Combinando tanto las nociones de detección y construcción del descriptor, encontramos el algoritmo *Speed Up Robust Features* o SURF (Bay, Ess, Tuytelaars, & Gool, 2008). SURF consta de tres etapas: detección de puntos de interés, descripción del vecindario local de los puntos de interés y correspondencia.

Este algoritmo extrae algunos puntos clave y proporciona un descriptor único para una imagen.

Se puede extraer un conjunto de puntos clave y descriptores SURF de una imagen y luego usarlos más tarde para comparar con otras imágenes y determinar si existen objetos comunes en ambas.

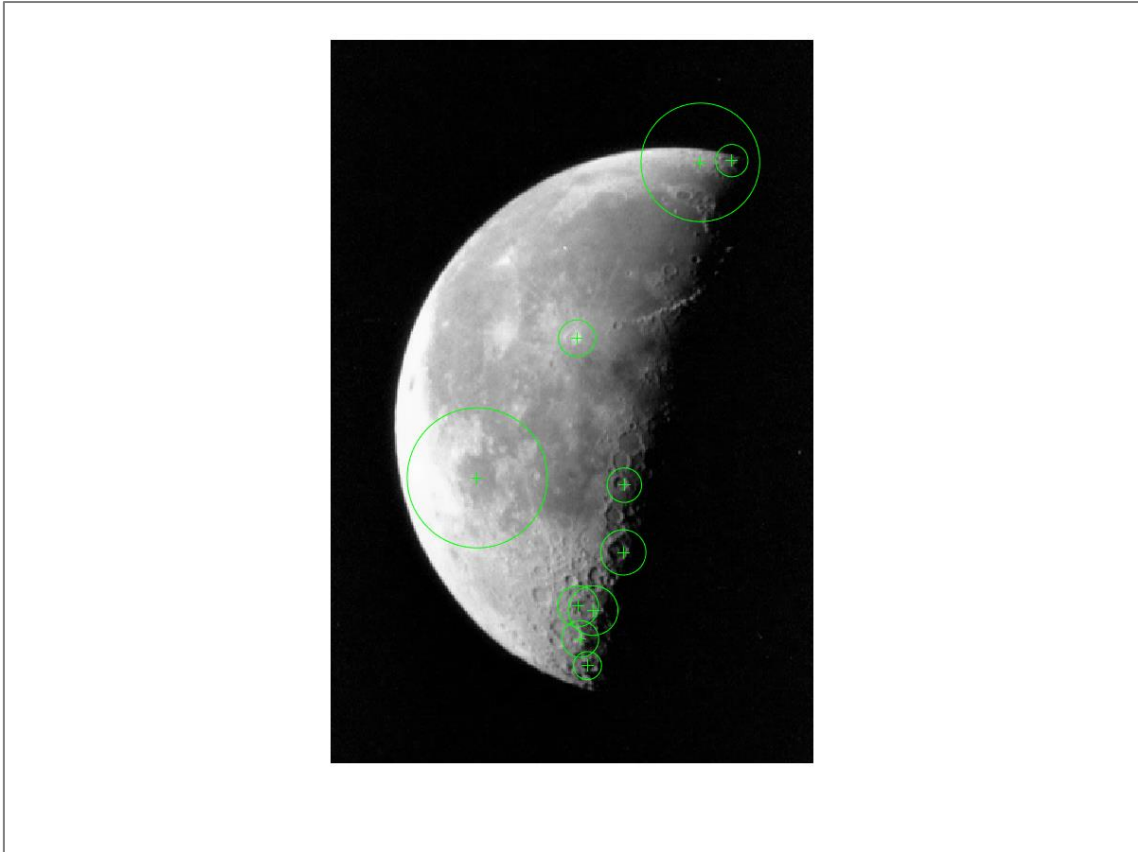
SURF utiliza una representación de imagen intermedia llamada Imagen Integral, que se calcula a partir de la imagen de entrada y se utiliza para acelerar los cálculos en cualquier área rectangular. Se forma sumando los valores de píxel de coordenadas  $x$ ,  $y$  desde el origen hasta el final de la imagen. Aunque pueda parecer que el tiempo de cálculo sea elevado, solo es preciso recorrer cada imagen una única vez. Además, el método es invariante a cambios de tamaño por lo que también es útil al emplearse en imágenes de grandes dimensiones.

El detector SURF se basa en el determinante de la matriz de Hesse. Los fundamentos de SURF residen en cómo se distribuyen las intensidades de los píxeles dentro de una vecindad dependiente de la escala de cada punto de interés.

La detección de objetos utilizando SURF es invariante a cambios de escala y rotación, por lo que tiene gran aplicabilidad. Tampoco requiere fases de entrenamiento, no obstante el tiempo requerido para las comparaciones entre distintas imágenes es un tanto considerable.

Dado que este método es invariable en rotación, es posible detectar objetos con éxito que aparezcan en cualquier orientación. Esto será particularmente útil en situaciones en las que se dispone de un modelo de objeto con una proyección dada, que puede aparecer con distintas orientaciones en otras imágenes.

Comparando los descriptores SURF de una imagen con los descriptores de SURF de otras imágenes, puede determinarse si un objeto aparece en ambas imágenes o no con una relativa precisión.



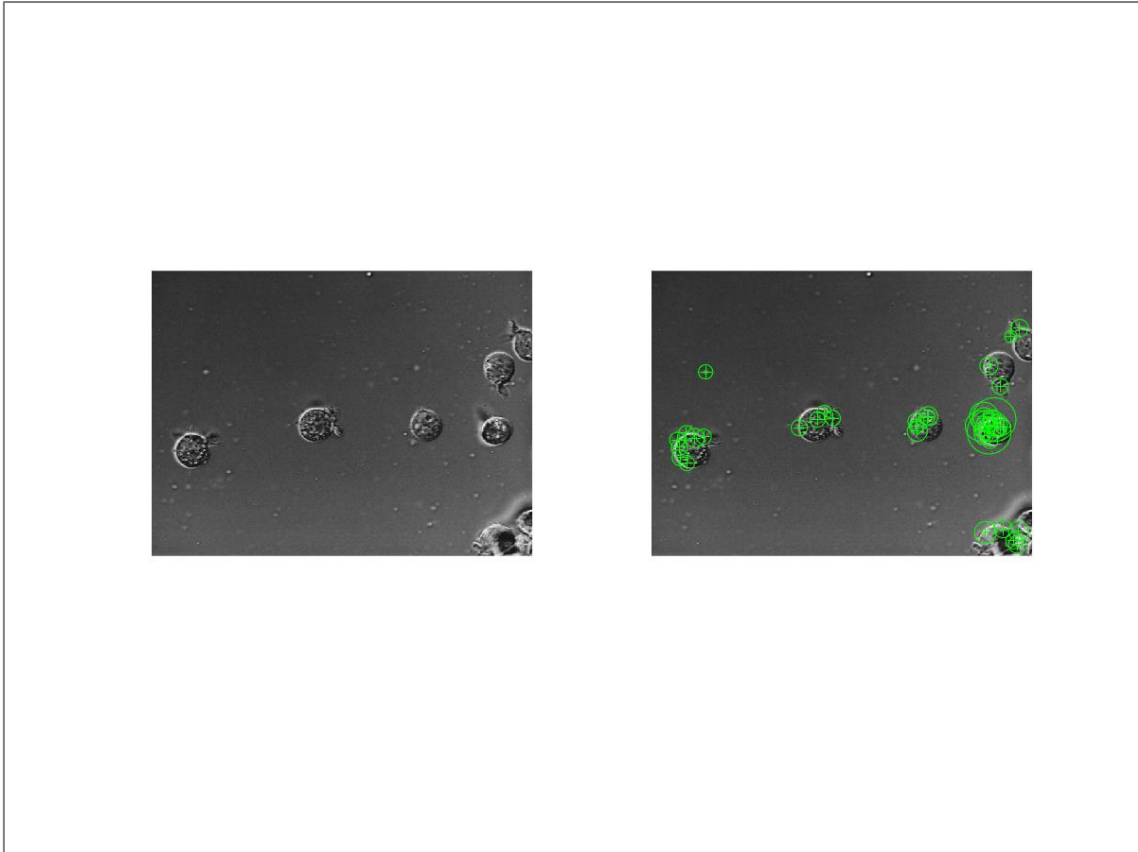
**Ilustración 2.2.3-3: Características detectadas por el Algoritmo SURF**

Otro algoritmo propone un nuevo descriptor de regiones inspirado en el sistema visual humano, concretamente la retina, es el denominado *Fast Retina Key-point*, FREAK, (Alahi, Ortiz, & Vandergheynst, 2012).

Se basa en calcular obtener una cascada de cadenas binarias mediante la comparación de las intensidades de la imagen respecto a un patrón de muestreo. El método FREAK es en general rápido y eficaz obteniendo características, con una menor carga de memoria y mayor robustez que otros métodos.

Es una alternativa muy competitiva para detectar regiones a partir de la condensación de puntos clave, y dado su buen rendimiento computacional suele ser una opción preferente para software embebido.

En la Ilustración 2.2.3-4, se muestran dos imágenes. A la izquierda una imagen a escala de grises obtenida a través de un microscopio electrónico en la que pueden apreciarse algunas bacterias. En la imagen de la derecha podemos ver los puntos detectados por FREAK. Los centroides de cada *cluster* de puntos podrían definir cada región.



**Ilustración 2.2.3-4: Características encontradas por el algoritmo por el Algoritmo FREAK**

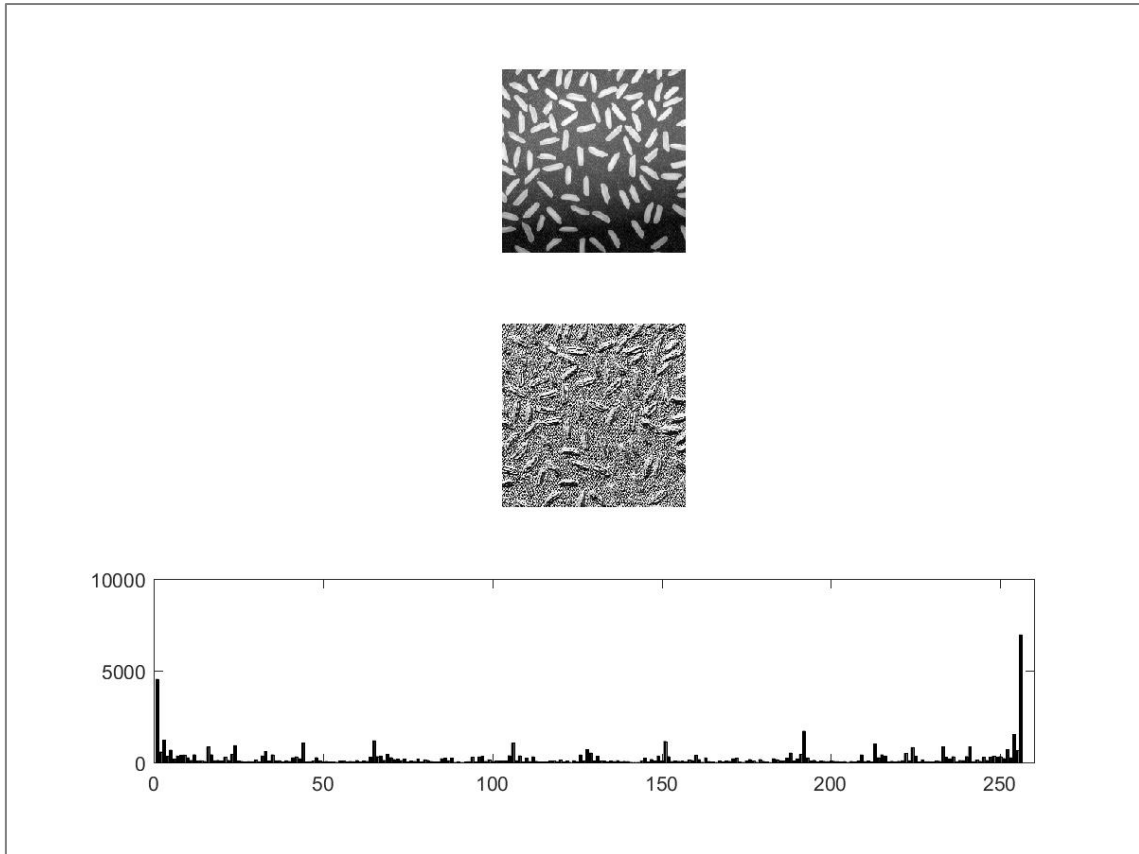
En último lugar es mencionado el operador *Local Binary Patterns* o LBP (Ojala, Pietikainen, & Maenpaa, 2002), cuya finalidad original era la detección de texturas. Este operador estudia el nivel de intensidad de gris del vecindario de cada píxel de la imagen para asignarle un valor entero.

Una vez calculado el patrón LBP de cada píxel, se construye un descriptor a nivel de imagen; cuya representación consiste en un vector o histograma definido por las ocurrencias de patrones LBP obtenidos.

Diferentes texturas dan lugar a diferentes regiones en una imagen digital; por lo que a ésta se le podría aplicar una segmentación de regiones basada en el Operador LBP.

En la sección 3.1.1 se realiza una descripción detallada del Operador LBP en términos procedimentales y matemáticos, por lo que no se profundizará más en la lógica del operador en esta sección. En la sección 2.2.4 se mencionan y resumen brevemente varios estudios recientes sobre el Operador LBP, especialmente sus variantes.

En la Ilustración 2.2.3-5 se muestran; una imagen normal, la imagen resultante de aplicar el Operador LBP a la primera imagen, y el histograma LBP.



**Ilustración 2.2.3-5: Descriptor LBP y su aplicación a una imagen**

### 2.2.4. Operador LBP

El estudio de referencia antiguo con más impacto es (Ojala, Pietikainen, & Maenpaa, 2002), en esencia porque incluyen la noción de patrones uniformes que son aquellos cuyo bit string posee como máximo dos transiciones de 0 a 1 o viceversa.

En realidad el operador LBP básico ya había sido propuesto por los mismos autores con anterioridad, pero es relativamente complicado encontrar referencias más antiguas publicadas en revistas de investigación, ya que en los años 90 del siglo XX este método era considerado *ad-hoc* por no disponer de un sólido fundamento teórico.

A partir de Ojala, con la innovadora consideración de los patrones uniformes y los buenos resultados en el estudio para la clasificación de texturas respecto a las técnicas contemporáneas, el operador ganó un creciente mayor interés.

El operador LBP ha continuado siendo motivo de estudio en los últimos años, desde distintas perspectivas por diversos motivos de investigación.

Destacan sin duda dos publicaciones recientes en relación a la escritura de este texto. La primera (Brahnam, Jain, Nanni, Lumini, & others, 2014) es un libro completamente dedicado al operador LBP, sus variantes y el problema de la clasificación de texturas en imágenes digitales en general.

Es necesario parar a hacer una mención especial al análisis que se incluye en el libro acerca de los histogramas de patrones de texturas equivalentes, así como de texturas que podrían ser consideradas como un patrón muy similar según qué operador fuese aplicado para la codificación de las características de esas texturas.

El texto de Brahnam también incluye una revisión de las distintas aplicaciones que se le ha dado al operador en los últimos años y por citar algunas de ellas, en el libro referencian ejemplos de análisis de imágenes faciales, aplicaciones biométricas, análisis de imágenes clínicas, análisis de actividad, eventos y movimiento u obtención de *frames* basada en contenido de bases de datos de imágenes o vídeos.

En la obra se incluye un capítulo dedicado a revisar las variantes del operador LBP. La primera que remarcamos es *Local Ternary Patterns* (LTP).

En esencia, el LTP introduce un umbral a la hora de realizar el cálculo de diferencias de intensidades entre el píxel central y su vecindario: en lugar de utilizar el valor de intensidad del píxel central, emplea dicho umbral. La razón de esta modificación al operador LBP es su supuesta sensibilidad al ruido uniforme, como podría ser el provocado por un filtro gaussiano.

Es referenciado un estudio dedicado a la clasificación de imágenes del universo microscópico bacteriano donde esta variante del operador junto a un sistema de aprendizaje basado en el Análisis de Componentes Principales tiene un éxito notable respecto a otras técnicas que tratan de solucionar el mismo problema.

Otra alternativa del operador LBP es *Fuzzy Local Binary Patterns* (FBP) cuyo objetivo es lidiar con la sensibilidad del operador original a determinados tipos de ruido. La idea de introducir ciertos criterios de lógica borrosa al operador original pretende resolver las pequeñas variaciones de intensidad que produce el ruido, siendo dicho ruido o bien artefactos u otros elementos o bien ruido introducido artificialmente.

En esencia, FPB embebe la lógica borrosa a la hora de determinar el bit string que define el patrón LBP de un píxel. En lugar de usar el conjunto binario, el bit string se define por conjuntos borrosos; a cada valor del bit string le acompaña una probabilidad de ser 0 y otra probabilidad de ser 1. La pertenencia a un conjunto u otro se resuelve finalmente de acuerdo a un umbral y la media de intensidades del vecindario.

La última sección del libro de Brahnam incluye algunos estudios dedicados a la aplicación del operador LBP y sus nuevas variantes enmarcados en el ámbito del reconocimiento de rostros.

La segunda publicación reciente destacable, también por su carácter integrador y holístico en relación a los orígenes y avances sobre el estudio del operador LBP es (Pietikäinen &

Zhao, 2015). Pietikäinen y sus colaboradores incluyen una revisión histórica del LBP desde sus orígenes hasta la fecha en que es publicado el texto.

Además, discuten algunas de las variantes que han ido apareciendo a lo largo de los años, como el *Completed Local Binary Patterns* (CLBP), una propuesta en la cual cada diferencia de intensidad entre el pixel central y cada miembro de su vecindario es calculada considerando una magnitud o peso de cada uno de los vecinos, además de la clásica del LBP.

Para calcular dichos pesos, es introducida la noción del signo, que depende de qué tipo de diferencia de intensidad exista entre el pixel central y su vecindario (si el centro es más o menos brillante) y una media de las intensidades del vecindario.

De este modo logran incorporar a la representación del descriptor información procedente del vecindario en sí en lugar de solo de la relación del píxel central respecto a su vecindario.

Otra variante mencionada en el recopilatorio de Pietikäinen es el denominado *Volume Local Binary Pattern* o sencillamente VLBP. Esta aproximación considera texturas como conjuntos de volúmenes  $(X, Y, T)$ , donde  $X$  e  $Y$  denotan coordenadas espaciales y  $T$  el índice del *frame*.

El vecindario de un píxel pues, es definido en un espacio tridimensional. Esto implica la obtención de tres histogramas LBP, en base a pares del conjunt  $(X, Y, T)$ .

Se señala la aplicación de este método para el reconocimiento de expresiones faciales, de género, lectura a partir de la forma de los labios y reconocimiento de acciones.

Otra variante discutida es la *Local Binary Gabor Patterns* (LBGP). Consiste simplemente en aplicar diferentes filtros de Gabor, a modo de preprocesamiento, a una imagen para posteriormente calcular los diferentes histogramas LBP de cada imagen filtrada y concatenar dichos histogramas en un único descriptor, con la consiguiente desventaja de introducir una excesivamente alta dimensionalidad al mismo.

Se identifican en texto muchas otras variantes, categorizadas en base a cómo o cuándo se introducen dichas variaciones, pudiendo ser: durante el preprocesamiento, el modo de definir la topología del vecindario y el muestreo, aplicando umbralizaciones o introduciendo modificaciones en la codificación del descriptor, la consideración de imágenes a color, la búsqueda de tolerancia a distintos tipos de ruido, la construcción del descriptor combinando la información local del mismo con información global; así como también numerosas hibridaciones del operador distintas con técnicas de aprendizaje o *Machine Learning*.



En resumen, puede decirse que la simplicidad y eficiencia computacional del operador LBP ha dado lugar a muchas variantes, evoluciones e hibridaciones de este mecanismo con otras técnicas, permitiendo resolver nuevos problemas, o resolver con un mejor rendimiento en términos de precisión problemas para los cuales ya existían otras propuestas para su resolución.

También es de especial interés los estudios señalados por Pietikäinen y compañía relativos a la gestión de la rotación y cambio de escala cometidos en distintos estudios. Tras indicar que la invarianza a rotación ya fue introducida en el operador, para clasificación de texturas, por Ojala en (Ojala, Pietikainen, & Maenpaa, 2002), son mencionadas otras aproximaciones, como *Adaptive Local Binary Pattern*, ALBP, que incorpora información direccional en el descriptor para problemas de clasificación de texturas.

En cuanto al cambio de escala e invarianza a rotación integrados, se incluye una referencia donde el vecindario del píxel es definido como un *texton* autoadaptativo, dando lugar a un histograma LBP circular.

Además de estas dos publicaciones que han sido destacadas con mayor profundidad, existen otras publicaciones recientes más focalizadas en contextos más concretos que también merece la pena que sean mencionadas y descritas brevemente.

En el estudio de Murala y sus colaboradores (Murala & Wu, 2014), los autores proponen el método que denominan *Local Mesh Patterns*, LMeP que viene a estar inspirado por el LBP y validan contrastando los resultados respecto al propio operador LBP, en su forma uniforme.

En realidad LMeP podría considerarse otra variante más del LBP, si se presta atención a la implementación, puede observarse que se introduce una variación que consiste en considerar información del vecindario y aplicar una umbralización a la hora de definir el patrón LBP, dando lugar a un patrón alternativo que incorpora cierta información del vecindario y además resulta en una reducción de los posibles patrones obtenidos.

La motivación de la introducción de esta aproximación es reducir la dimensión del histograma del descriptor.

En los resultados experimentales indican una mejora del rendimiento de su LMeP, que achacan a una mayor capacidad de capturar información sobre los bordes de las texturas. Dichos experimentos se centran en imágenes de resonancia magnética cerebrales, concretamente del plano cortical.

De manera adicional, también incluyen otro experimento aplicando un preprocesamiento con filtros de Gabor, dando lugar a lo que denominan GLMeP. De nuevo, la idea

subyacente es generar varias imágenes derivadas de aplicar el filtro de Gabor en base a distintas parametrizaciones del mismo con el fin de obtener varios descriptores LMeP diferentes para posteriormente concatenarlos en un descriptor GLMeP.

En esta ocasión los resultados son contrastados con el GLBP.

Otra reciente variante del LBP la aporta Qi en (Qi, y otros, 2014), denominada *Pairwise Rotation Invariant Local Binary Pattern*, PRI-CoLBP. Está totalmente orientada a dotar al LBP de invariabilidad a cambio de escala e invariabilidad a rotación, con un enfoque innovador.

Para ello recurren a la coocurrencia de características, en el sentido de observar las coocurrencias en parejas de características en lugar de atender a ocurrencias de dos características por separado.

Esto se traduce en que en lugar de estudiar el vecindario de un único píxel para determinar un patrón, son estudiados píxeles dos a dos y la matriz de coocurrencias de los mismos para determinar un patrón “CoLBP”.

Así es posible codificar en el patrón una estructura más compleja, e incorporar a dicha codificación un ángulo, entre los dos píxeles estudiados y sus vecindarios, por lo que no importa cómo rote la imagen: los vecindarios podrán variar pero el ángulo siempre sería constante.

Para lograr alcanzar la invarianza a cambio de escala y rotación conjuntamente, es preciso incorporar más información local, por tanto estudian más matrices de coocurrencias para tres escalas y dos orientaciones. En total serían 6 matrices de coocurrencias; si además es considerado el color, serían un total de 18, obteniéndose un vector de características con un gran número de dimensiones.

No obstante, señalan que es posible aplicar técnicas de reducción de la dimensionalidad habituales como el Análisis de Componentes Principales (PCA).

En los resultados experimentales demuestran una mejora de la tolerancia a la rotación según son añadidas matrices de coocurrencia adicionales, así como una mejora en imágenes a color.

Finalmente contrastan su método respecto a estudios sobre clasificación de tejidos y reconocimiento de materiales, sobre tres *datasets* de imágenes distintos, obteniendo como mínimo los mismos resultados que en dichos estudios.

Chen propone la utilización del LBP junto a los *Depth Motion Maps* DMM (Chen, Jafari, & Kehtarnavaz, 2015) con el objetivo de reconocer acciones humanas, problema de interés en áreas de vigilancia inteligente, asistencia o juegos, entre otros.

Los DMM contienen per sé información estructural tridimensional; la idea que plantean los investigadores es explotar esa información codificando bloques solapados de las distintas proyecciones de cada *frame* del DMM mediante el descriptor LBP, para posteriormente concatenarlos obteniendo un único descriptor.

Pero no solamente es concatenados cada histograma LBP, sino que hacen dos concatenaciones diferentes: una simple a nivel de característica y otra más compleja a nivel de decisión. Con esta aproximación, lo que realmente aportan son dos tipos de clasificador distintos guiados por el mismo algoritmo de decisión.

En los resultados comparan su método respecto a varios métodos planteados en otras investigaciones para la identificación de acciones humanas en un espacio tridimensional, obteniendo resultados competitivos a nivel de precisión.

También incorporan una reflexión sobre la eficiencia computacional de su método, resaltando precisamente esta propiedad.

Dentro del marco metodológico para resolver problemas de clasificación en imágenes hiperespectrales, Li y Chen (Li, Chen, Su, & Du, 2015) y demás colaboradores incorporan la utilización del operador LBP para la codificación de información local de las imágenes, filtros Gabor para incorporar información global de las texturas, características espectrales y una red neuronal tipo Elman (ELM) para realizar las labores de clasificación de las texturas.

Conociendo las clases para un conjunto de prueba, entrenan la red de acuerdo a las texturas local, global e hiperespectral. Validan su metodología comparándola con otras propuestas en el ámbito de la clasificación de imágenes hiperespectrales, concluyendo que las representaciones del LBP son efectivas para la extracción de características en este tipo de imágenes, debido a que codifican configuración de texturas y estructuración a nivel local

## **2.3. Metaheurísticas y Operadores de Visión**

En el dominio de la Inteligencia Artificial, es empleado el término heurística habitualmente para referenciar la aplicación de conocimiento del dominio a un problema específico.

En contraposición, la noción de Metaheurística se orienta a la aplicación de estrategias holísticas para resolver problemas computacionales de carácter general; es decir, una Metaheurística sería aplicable a diferentes tipos de problema indistintamente.

Los algoritmos evolutivos pueden emplearse como Metaheurísticas de optimización y/o búsqueda, así como otros algoritmos basados en poblaciones.

Cuando concretamente son empleados Algoritmos Evolutivos para resolver problemas típicos del campo de la Visión Artificial, numerosos autores describen esta aproximación como Visión Artificial Evolutiva (Cagnoni, Lutton, & Olague, 2007).

A continuación, se ofrece una breve descripción de diferentes estudios e investigaciones donde son propuestas, aplicadas y evaluadas distintas Metaheurísticas sobre diferentes problemas de Visión Artificial.

En el estudio realizado por Lutton (Lutton & Martinez, 1994), se investiga la utilización de un Algoritmo Genético como alternativa a la utilización de la Transformada de Hough para la extracción de contornos. Una explicación sobre la Transformada de Hough y su aplicación a imágenes digitales, está disponible en (Duda & Hart, 1972).

Es destacable del estudio del Lutton el planteamiento acerca de la representación más adecuada de los individuos para localizar segmentos, círculos o elipses. La población es iniciada aleatoriamente en base al tipo de elementos que se desea encontrar en la imagen de acuerdo a los modelos de representación establecidos.

En los resultados concluyen que para cierto tipo de figuras, el método planteado es complementario a la Transformada de Hough. No obstante, el esquema propuesto por los autores fue lo suficientemente innovador y sirvió de inspiración a otros investigadores para estudiar la aplicación de Algoritmos Genéticos en distintos problemas de Visión Artificial.

Hace ya algunos años, en (Swets, Punch, & Weng, 1995) se plantea una metodología para la localización de objetos en imágenes basada en segmentación y clasificación, donde dichas labores de segmentación y clasificación son encomendadas a un Algoritmo Genético; la misión de este motor de búsqueda no es solo encontrar objetos sino también clasificarlos.

Para ello, definen un cromosoma donde subconjuntos de bits definen; clase, traslación desde el centro de la imagen, altura y anchura.

El principal problema de la representación es el elevado número de combinaciones no permitidas. Esto supone que dentro del espacio de búsqueda existe gran cantidad de soluciones no válidas, como por ejemplo clases que no existen. Por esta razón, la función de reparación empleada para poder asignar un coste adecuado a cada individuo es un tanto enrevesada.

Los experimentos consisten en la aplicación del método a tres imágenes cuya complejidad aumenta por el número de clases y el tipo de fondo de la escena.

En las conclusiones manifiestan la necesidad de realizar numerosas reconsideraciones para hacer que el método sea más generalizable.

En otra investigación (Centeno, Lopes, Felisberto, & Ramos, 2005) es señalado que los criterios de similitud para determinar si un objeto está o no presente en una escena presentan una dependencia considerable sobre sí el objeto concuerda con el template tras aplicarle transformaciones afines, es decir, si hay una variación de la escala o de la inclinación.

Debido al hecho señalado, plantean como Metaheurística la utilización de un Algoritmo Genético cuyo genotipo son cinco dimensiones que representan: las coordenadas del píxel central del individuo, un factor de escalado, un factor de rotación y un umbral.

El motivo de incluir un umbral se debe a que en la decodificación el primer paso es binarizar la imagen individuo según el valor indicado por el cromosoma umbral (el template ya está binarizado).

Emplean el Error Cuadrático Medio como criterio de similitud, pero enfocado a la comparación directa entre la matriz imagen individuo y la matriz de la imagen del template.

En los experimentos realizados en el estudio, puede apreciarse que las imágenes no incluyen texturas heterogéneas, en realidad todas las escenas son figuras de ajedrez y el template un alfil.

Como conclusión, manifiestan que el método planteado encuentra soluciones cercanas al óptimo global para la mayoría de los casos y que el coste computacional es bajo.

Trujillo y Olague (Trujillo & Olague, 2006) plantean la detección de puntos de interés desde el enfoque de un problema de optimización resoluble mediante el uso de Programación Genética, empleando la tasa de repetición—*Repeatability Rate*—junto a una medida de entropía relacionada con la distribución de puntos sobre la imagen, como función fitness del algoritmo genético.

En esta ocasión lo que se busca no es aplicar la Metaheurística para encontrar un objeto en una imagen, sino encontrar un Operador de Visión que facilite analizar las imágenes y representarlas mediante un descriptor que finalmente contribuya a localizar objetos en imágenes.

En dicho operador “sintético” buscan determinadas propiedades: la compactación de gran cantidad de información, que el vecindario de cada punto de interés sea lo suficientemente poco frecuente en la imagen, estabilidad ante cambios de escala y rotación y una cierta distribución entre los puntos de interés localizados sobre la imagen escena.

La función de coste, dada la naturaleza del estudio, se define en términos de tasa de repetición, del individuo respecto a una serie de imágenes. Es preciso señalar, que inicialmente las posibles soluciones son constituidas—aplicando operadores biológicos

de selección, recombinación y mutación—sobre un conjunto de imágenes de entrenamiento y posteriormente se valida su coste.

Una vez obtenidos una serie de posibles operadores sintéticos, estos son comparados experimentalmente respecto a operadores tradicionales sobre un *dataset* de imágenes de *benchmarking*.

Como conclusión, presentan el descubrimiento de una variante del Operador DET—DET de “Determinante”—, demostrando así la capacidad del método para encontrar nuevos Operadores de Visión o mejoras de operadores ya existentes.

Silva y Lopes investigaron introducir modificaciones al *Compact Genetic Algorithm*, CGA, (Silva, Lopes, & Erig, 2008) y denominan a su aproximación “emCGA” con el objetivo de validar su rendimiento en coste computacional y de almacenamiento en memoria.

Para ello eligen el reconocimiento de objetos en imágenes digitales, un problema típico donde el coste computacional es elevado.

Es destacable de este método, su propuesta para definir el modelo de objeto, dado que dicho modelo es tridimensional: esto presenta una ventaja muy destacable ya que la correspondencia que se analiza entre un hipotético individuo del plano escena respecto al modelo se basa en desplazamientos por los planos de los ejes del modelo y rotación en los tres planos del mismo.

No obstante en los experimentos limitan los desplazamientos solo a los ejes x e y y una posible rotación en dicho plano, por lo que los individuos quedan representados por un vector de tres dimensiones.

Como medida de similitud entre template e individuo emplean el Error Absoluto de las matrices imagen.

Los experimentos consisten en comparar el número de evaluaciones del método propuesto en contraposición a las evaluaciones del algoritmo *Exhaustive Algorithm Search* para dos escenas y dos modelos, constatando la superioridad en este caso de su aproximación.

El algoritmo PSO, *Particle Swarm Optimization*, es la elección de los investigadores en el estudio (Perlin, Lopes, & Centeno, 2008) para la búsqueda de objetos y patrones en imágenes digitales.

Los individuos son definidos por una cuaterna donde las dos primeras dimensiones corresponden a las coordenadas del centro del individuo y las dos últimas representan factor de escalado y rotación.

Como medida de similitud entre candidatos y template, se sirven del Error Absoluto de las diferencias entre los valores de intensidad de los píxeles del template y el individuo en niveles de gris.

Plantean la misma dinámica para imágenes a color, promediando el error absoluto de las diferencias de cada canal.

Los experimentos presentados se centran en comprobar la tolerancia de la función fitness a cambios de escala y de rotación, en comprobar la tolerancia a ruido gaussiano, comprobar la efectividad cuando existe oclusión, y por último cuando la escena puede presentar numerosos óptimos locales (buscar un rostro concreto en una fotografía donde aparecen varias personas).

En resumen, los experimentos se orientan a cómo sortear las dificultades del problema de reconocimiento de objetos referentes a cambios de escala, rotación, ruido y oclusión.

Aunque obtienen unos resultados aceptables en términos de precisión, los propios autores señalan como trabajo futuro de investigación la utilización de otras funciones *fitness*.

Chidambaram y Lopes (Chidambaram & Lopes, 2009) proponen la utilización del algoritmo *Artificial Bee Colony*, ABC, como motor de búsqueda para la detección de objetos en imágenes—a color—. La idea básica de este algoritmo es disponer de una población con dos tipos de castas según su función: “abejas exploradoras” y “abejas obreras”.

Las exploradoras viajan por el espacio de soluciones buscando candidatos a solución en base a un coste, las obreras en cambio se concentran en explotar el espacio cercano a soluciones lo suficientemente prometedoras detectadas por las exploradoras.

Emplean una representación de los miembros de la población de la colonia de cuatro dimensiones, donde las dos últimas representan respectivamente un factor de escalado y un ángulo de rotación.

Para determinar la calidad de una posible solución, como criterio de similitud entre el objeto a identificar y dicha posible solución comparan los histogramas RGB de cada imagen en base al Error Cuadrático Medio.

En los experimentos, utilizan dos templates distintos para localizar objetos parecidos en la escena. Comparan los resultados con otros algoritmos destacando haber reducido el promedio del número de evaluaciones para alcanzar la solución.

Una alternativa al algoritmo PSO, es el algoritmo *Bat Algorithm*, BA: Zhang y su equipo en (Zhang & Wang, 2012) emplean una variante del BA a la que es añadida una etapa de mutación, y denominan a dicha variante BAM, donde la “M” es por *Mutation*. Contrastan

su Metaheurística respecto a los algoritmos BA, *Differential Evolution* (DE) y el *Simple Genetic Algorithm* (SGA) en los experimentos del estudio.

Atendiendo a la función *fitness* que presentan, se aprecia que como criterio de similitud no explotan ningún operador de Visión Artificial, sino que directamente calculan la distancia Euclídea entre las matrices imagen del template y cada miembro de la población.

Los experimentos definen una única escena y un template que pertenece a la escena, donde las dimensiones del template son menores que las de la escena, sin contemplar cambios de escala ni rotación.

En los resultados se aprecia una mejora del rendimiento del algoritmo en cuanto al tiempo de ejecución.

Una muy buena definición del marco de la aplicación de Metaheurísticas al problema de la detección de objetos en imágenes digitales es proporcionada por (Ugolotti, y otros, 2013): “*dado un modelo paramétrico de un objeto a detectar en una imagen; una función que representa la similitud entre un modelo y una región de la imagen, es maximizada*”.

Concretamente Uglotti en este estudio apuesta por usar como motor de búsqueda del óptimo global de dicha función de similitud los algoritmos *Particle Swarm Optimization* y *Differential Evolution*, también con la finalidad de hacer una comparativa entre ambas Metaheurísticas. Escogen PSO y DE por su capacidad para paralelizar cálculos.

Enfocan la comparativa a un problema desde dos perspectivas. La primera sería tratar de estimar la pose de una figura humana en una imagen. La definición paramétrica de dichas poses, presenta un elevado número de dimensiones y está definida en base a conocimiento del dominio. Los algoritmos persiguen encontrar ciertas poses en un conjunto de imágenes.

La segunda perspectiva se orienta a otro tipo de problema, la localización del hipocampo en planos sinusoidales (imágenes histológicas) de cerebros de ratones. El modelo paramétrico planteado se sirve dos conjuntos de puntos: uno de puntos del borde interior del hipocampo, otro de puntos exteriores y las posiciones relativas de parejas de puntos de cada conjunto.

La función objetivo persigue minimizar las intensidades de los puntos que pertenecen al conjunto de puntos interiores, maximizar las intensidades de los puntos exteriores así como preservar las posiciones relativas del modelo.

Los experimentos son ejecutados sobre un conjunto de *frames*, uno para la estimación de poses y otro para la detección del borde del hipocampo.



Finalmente los resultados de dichos experimentos demuestran que con un rendimiento similar, un algoritmo obtiene mejores resultados en un problema y el otro en el otro.

Para la detección de matrículas, Smara y sus colaboradores (Smara & Khalefah, 2014) diseñan un *framework* que se basa en la utilización de un Algoritmo Genético junto un análisis de regiones conectadas—*Connected Component Analysis*—para guiar la búsqueda en escenas donde aparecen vehículos cuyas matrículas presentan grandes diferencias entre sí: por ser de distintos países, aparecer en escalas diferentes, tener distinto color o iluminación, etc.

Dicho *framework* consta de una serie de etapas de preprocesamiento de las escenas, en las cuales se aplican diversos filtros: umbralización, transformaciones morfológicas, análisis de regiones conectadas y otros para finalmente permitir al genético realizar la búsqueda de la matrícula.

El genotipo de consta de siete genes codificados con números enteros, que representan los índices de los objetos que han sido detectados en la escena tras aplicar las etapas de procesamiento. De cada objeto, además de su índice se conocen las coordenadas de la ventana que encapsula al objeto, así como la altura y anchura de dicha ventana. Entienden ventana como subimagen de la escena.

La función *fitness* evalúa si la composición de objetos definidos por el genotipo, presentan una disposición geométrica similar a la que presentarían los números de una matrícula, con sus respectivas ventanas encapsulando dichos números.

La evaluación se enfoca en la relación de las posiciones de los objetos, así como la relación de los tamaños.

Para los experimentos emplean una gran cantidad de escenas donde aparecen distintos vehículos de diferentes países, obteniendo elevadas tasas de éxito en la mayoría de ellas.

Para finalizar es revisado el estudio (Paulinas & Ušinskas, 2015), donde los autores hacen una reflexión de los usos que se le ha dado en los últimos años a los Algoritmos Genéticos para resolver distintos problemas de Visión Artificial.

La investigación se centra en las aplicaciones de los genéticos para la mejora de la calidad de las imágenes y para tareas de segmentación.

En relación a esto último, se referencian varios estudios donde encontramos Algoritmos Genéticos ejerciendo el rol de optimizador de parámetros para varias técnicas de segmentación diferentes.

Las conclusiones planteadas por Paulinas son la afirmación de que cada aproximación es única en el sentido de que, el éxito de la segmentación depende por completo de las

elecciones de diseño típicas que se dan a la hora de utilizar un Algoritmo Genético para cualquier problema: esquema del cromosoma o genotipo, tipos de operadores biológicos utilizados, configuración de parámetros del algoritmo así como la función fitness, que en el caso de servir para validar la bondad de una segmentación, puede ser especialmente compleja.

Se ha revisado en el Estado del Arte de esta memoria algunos tipos de Operadores de Visión, haciendo especial énfasis en el Operador LBP y también se han revisado algunos estudios orientados a la aplicación de Metaheurísticas a problemas de Visión Artificial, prestando mayor atención a aquellos pertenecientes al ámbito de la detección o reconocimiento de objetos.

Algunos de estos estudios se han servido de las ventajas de aplicar Operadores de Visión de un modo u otro, pero dicha aproximación no goza de una atención generalizada.

En el próximo capítulo se describe cómo aprovechar la aplicación del Operador LBP junto a la Metaheurística *Differential Evolution* con el fin de localizar objetos en escenas dado un *template* o modelo.



## **CAPÍTULO 3. DESCRIPCIÓN DE LA METODOLOGÍA**

---

## 3.1. Conocimiento Previo

En esta sección se describen detalladamente los fundamentos del Operador LBP y del algoritmo *Differential Evolution*.

### 3.1.1. Descripción Operador LBP

El operador LBP, por sus siglas en inglés *Local Binary Pattern*, es un descriptor que asigna un identificador binario a cada píxel de una matriz imagen mediante la comparación de los niveles de intensidad de cada píxel respecto su vecindario.

Debido a su poder discriminativo y simplicidad computacional, el operador LBP se ha convertido en un enfoque popular en varias aplicaciones de diversa índole en Visión Artificial.

Las propiedades más importantes del operador LBP en aplicaciones del mundo real son fundamentalmente su capacidad para distinguir texturas y su robustez a las variaciones constantes de intensidad luminosa, por ejemplo, por cambios de iluminación.

Otra propiedad importante es su simplicidad computacional, que permite analizar imágenes con desafiantes configuraciones en tiempo real.

#### **Obtención del vector de características LBP**

En esta subsección se incluye una descripción de las etapas que sigue el operador para determinar el patrón LBP de un píxel así como la obtención del histograma, vector de características o descriptor LBP de una imagen.

La idea esencial de este operador es el análisis de la variación de niveles de intensidad entre cada píxel de una imagen  $I$  y el vecindario de píxeles de cada píxel de la imagen  $I$ , para obtener una imagen  $I'$  umbralizada donde cada píxel ha adoptado un valor en relación a dicha diferencia de intensidades.

Además, según cuál sea la distribución de la relación de diferencias de intensidad de los píxeles de la imagen  $I$  respecto a sus vecinos, se construye un vector de características LBP.

#### **Muestreo**

Dado un píxel en una matriz imagen, que no esté situado en los límites de dicha matriz, para el Operador LBP el vecindario se define de forma circular, en base a un número de vecinos y un radio, como en la Figura 3.1.1-1 donde hay 8 vecinos y el radio es 1.

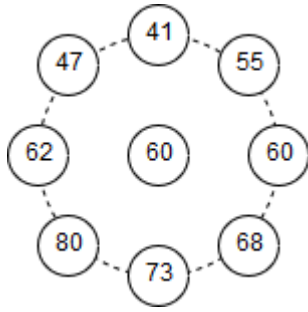


Figura 3.1.1-1: Vecindario de un píxel y sus niveles de intensidad luminosa

### Calculo de la diferencia de intensidades

Simplemente consiste en restarle a cada píxel del vecindario el valor de intensidad del píxel central, como paso previo o simultáneo a la binarización, como en la Figura 3.1.1-2:

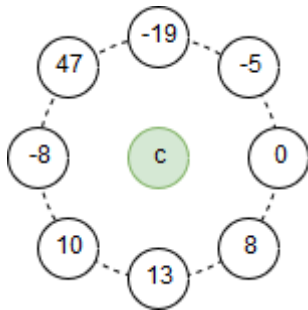


Figura 3.1.1-2: Diferencia de intensidades de un píxel central respecto a su vecindario

### Binarización de la diferencia de intensidades

Todos aquellos píxeles cuya diferencia de intensidad respecto al píxel central sean menor que cero, tomarán el valor binario cero. De forma complementaria todos aquellos píxeles del vecindario cuya diferencia de intensidad sea igual o mayor que cero, tomarán el valor 1, ejemplo en la Figura 3.1.1-3

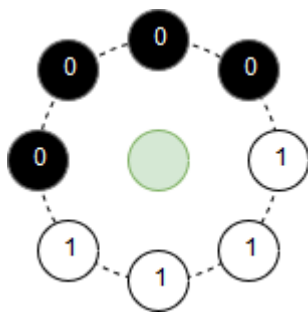


Figura 3.1.1-3: Vecindario binarizado

Comenzando con el píxel del vecindario situado a la derecha del píxel central, en sentido de las agujas del reloj, recorreremos el vecindario codificando un *bit string*, en este caso sería el 11110000.

### Asignación decimal

El patrón LBP de este píxel, se trata de la conversión a decimal del *bit string* obtenido en el paso anterior, recorriéndolo de manera invertida:

$$0*2^7 + 0*2^6 + 0*2^5 + 0*2^4 + 1*2^3 + 1*2^2 + 1*2^1 + 1*2^0 = 15$$

El patrón LBP de este píxel es 15.

### Inclusión en el vector de características

El bit string de longitud 8 suponen  $2^8$  posibles patrones, en el caso de este ejemplo el patrón es 15. En el vector de características LBP al que perteneciera este píxel, se sumaría una unidad a la posición 15 del vector. Nótese que  $2^8$  es la longitud máxima del vector de características LBP para una vecindad de tamaño 8, en el caso de que se contabilicen en bruto todos los posibles patrones.

### Diagrama del proceso

Los pasos descritos anteriormente pueden resumirse como el proceso que describe la Figura 3.1.1-4:

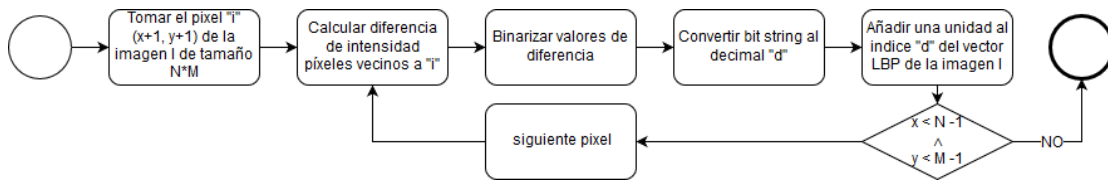


Figura 3.1.1-4: Proceso del Operador LBP

Nótese que es preciso hacer esta operación para cada píxel de la matriz imagen, que debe ser recorrida en orden. Las coordenadas de un píxel en una matriz imagen se asignan de izquierda derecha y de arriba a abajo como indica la siguiente Figura 3.1.1-5:

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)

(y, x)

Figura 3.1.1-5: Coordenadas de una matriz imagen

### **Modelo matemático del Operador LBP**

El patrón LBP de un píxel de coordenadas (y, x) viene dado por la Ecuación 3.1.1-1:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p$$

**Ecuación 3.1.1-1: Formulación del patrón LBP de un píxel**

Donde:

- $P$  es el tamaño del vecindario, el número de píxeles vecinos a considerar
- $R$  es el radio del vecindario.
- $p$  es el índice de píxel vecino
- $g_p$  es el valor de intensidad del píxel vecino de índice  $p$
- $g_c$  es el valor de intensidad del píxel central
- $s$  es una función cuya fórmula es indicada por la Ecuación 3.1.1-2:

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

**Ecuación 3.1.1-2: Binarización para la diferencia de intensidades de un vecindario**

Por otra parte, el cálculo del histograma LBP se expresa mediante la Ecuación 3.1.1-3:

$$H_I = \sum_{x=1}^{N-1} \sum_{y=1}^{M-1} LBP_{P,R}(i(x,y))$$

**Ecuación 3.1.1-3: Construcción del vector o descriptor LBP de una imagen**

Donde

- $N$  es la anchura de la imagen
- $M$  es la altura de la imagen
- $i$  es el píxel de coordenadas  $(x,y)$

### Parámetros del LBP

La naturaleza del vector de características LBP está sujeto a dos parámetros;  $P$  representa el tamaño del vecindario, es decir la cantidad de vecinos.  $R$  representa el radio. Dependiendo de estos parámetros, los píxeles vecinos cuya intensidad en nivel de gris se comparará con el píxel central serán un subconjunto de píxeles u otro, estarán más o menos cerca espacialmente del píxel central y también influirán en el tamaño del vector de características LBP de la imagen.



En la Figura 3.1.1-6 se plasman algunos ejemplos del histograma LBP resultado de aplicar el operador a una misma imagen empleando distintas combinaciones de R y P:

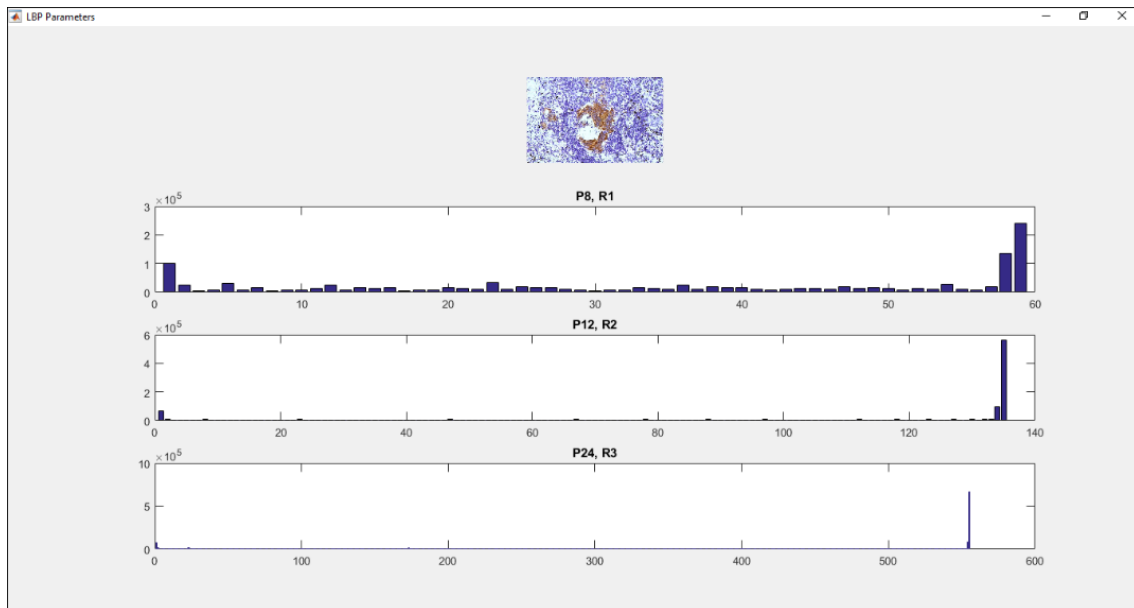


Figura 3.1.1-6: Descriptores LBP según los parámetros P y R

La cantidad de patrones LBP aumenta exponencialmente en base al tamaño del vecindario y el radio, tal y como puede apreciarse en la figura anterior. La ampliación del radio y del vecindario implica la obtención de un mayor número de posibles características. Nótese que en la figura la longitud de dichos vectores no coincide con el valor  $2^P$  debido a la utilización de una variante del LBP concreta. En la siguiente sección se explican dichas variantes.

## Variantes

En este apartado son descritas dos variantes clásicas de la implementación original del LBP. El procedimiento para la obtención del descriptor es el mismo que el descrito en este capítulo, la diferencia reside en que determinados conjuntos de patrones LBP serán agrupados de acuerdo a los criterios que se exponen a continuación.

### LBP uniforme

Esta variante es ampliamente utilizada en la literatura. La longitud del descriptor LBP uniforme queda reducida de acuerdo al agrupamiento de todos los patrones denominados no uniformes en una misma “categoría”.

Los patrones uniformes son aquellos cuyo *bit string* posee como máximo dos transiciones de 1 a 0 y/o de 0 a 1: 01000000 sería un patrón uniforme. Cada uno de los patrones uniformes tendrá definida su propia clase en el descriptor. De esta manera, la longitud del histograma queda reducida a 59, en lugar de tener longitud 256 como en el LBP original.

En términos de eficiencia, es evidente la mejora solo por el mero hecho de haber reducido el número de dimensiones del descriptor, como se aprecia en Figura 3.1.1-7

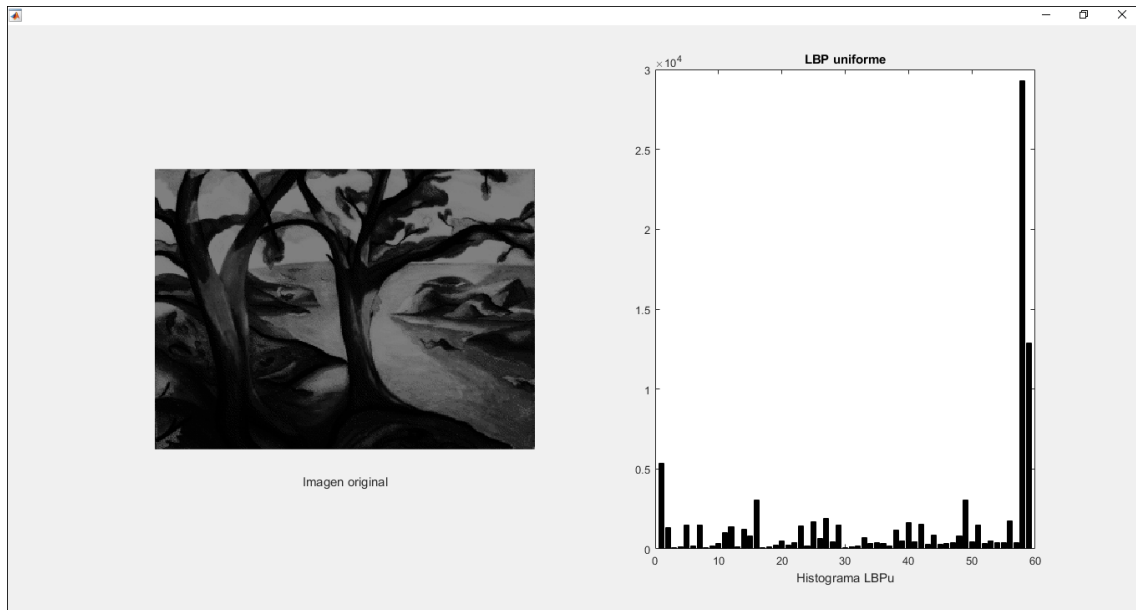


Figura 3.1.1-7: Ejemplo de descriptor LBP uniforme

En este estudio ha sido empleado el LBP uniforme en todos los experimentos ejecutados, pero por simplicidad nos referiremos a él simplemente como LBP.

### LBP invariante a rotación

Dado que durante el proceso mediante el que se obtiene el valor LBP de un píxel, hay una etapa en la que hay que recorrer un bit string, y dado que todos los *bit string* de todos los píxeles se recorren siempre en el mismo orden, se deduce que el operador LBP original no es invariante a rotación, como demuestra la comparación de Tabla 3.1.1-1.

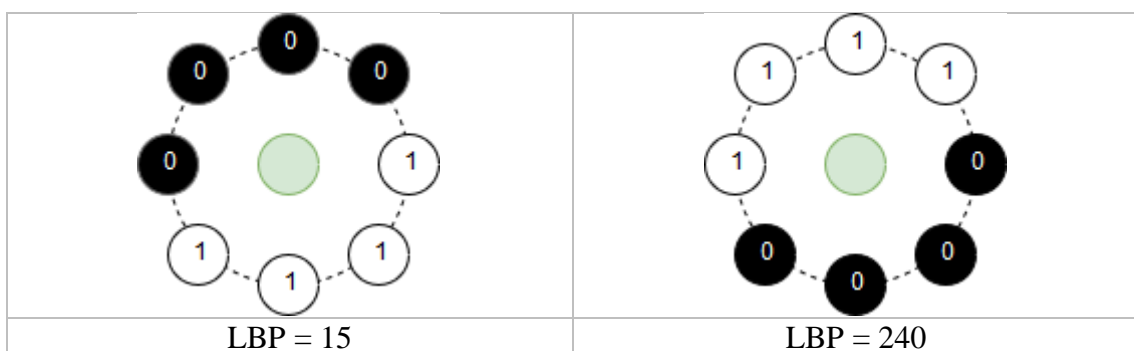


Tabla 3.1.1-1: Patrón LBP rotado

Para evitar este inconveniente, fue diseñado un mecanismo que además de generar el bit string del vecindario, analiza todos los bit strings derivados de las posibles rotaciones, lo que da lugar continuando con el ejemplo anterior, a 8 rotaciones y *bit string* posibles como ilustra la Figura 3.1.1-8.

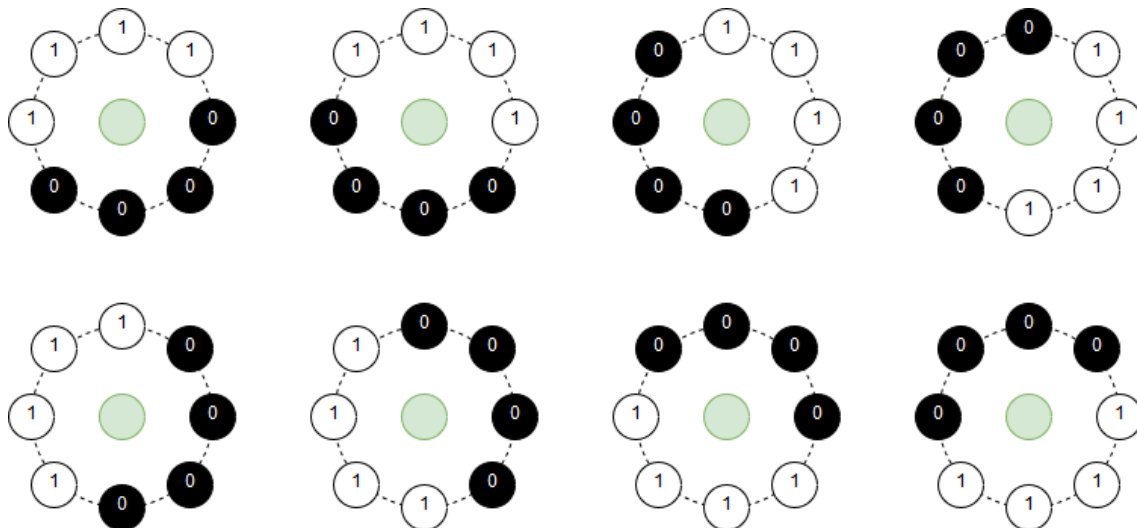


Figura 3.1.1-8: Rotaciones de un mismo patrón LBP

Calculando el LBP de estos ocho vecindarios/*bit string* se obtienen 8 patrones LBP, bien, pues el patrón cuyo valor sea el mínimo será elegido como patrón definitivo LBP para el píxel central analizado.

### 3.1.2. Differential Evolution

En esta sección se explica detalladamente el algoritmo básico de *Differential Evolution* (DE) Evolución Diferencial.

Los campos de la ciencia y la ingeniería, la toma de decisiones empresariales o la industria están todos repletos de problemas que requieren la aplicación del enfoque de la optimización. Dado que la mayoría de los problemas de optimización del mundo real parecen ser fundamental y prácticamente complejos de resolver analíticamente, la investigación en algoritmos que perfeccionen el proceso de optimización encontrando soluciones más aproximadas o soluciones viables en menos tiempo sigue siendo valiosa y activa.

Hoy en día, existen muchos algoritmos de optimización que funcionan utilizando técnicas de búsqueda basada en gradientes y heurísticas en contextos determinísticos y estocásticos. Con el fin de ampliar la aplicabilidad de técnicas de optimización a problemas más complejos, los principios naturales y físicos se mimetizan para desarrollar algoritmos de optimización robustos para este tipo de problemas, dando lugar a Metaheurísticas.

Los Algoritmos Evolutivos, *Simulated Annealing*, *Ant Colony Optimization*, *Memetic Algorithms*, *Particle Swarm Optimization* son algunos ejemplos de algoritmos de optimización orientados a problemas complejos, donde la función objetivo no es continua, o no es diferenciable, es multidimensional o está sometida a restricciones entre otras posibles dificultades.

Como rama de la Computación Evolutiva, *Differential Evolution* es un algoritmo relativamente simple y fácil de implementar que exhibe un buen rendimiento en términos de velocidad y eficacia encontrando soluciones aproximadas en problemas donde los parámetros de la función objetivo pertenecen al dominio de los números reales.

Dadas estas características, para el problema del *Template Matching* se escogió *Differential Evolution* como método de optimización y mecanismo primario de búsqueda de la solución.

*Differential Evolution* es una herramienta de diseño de gran utilidad a la cual se le puede dar numerosas aplicaciones prácticas. DE se ha utilizado en varias aplicaciones de ciencias e ingeniería para descubrir soluciones eficaces a problemas casi intratables sin apelar a conocimientos expertos o algoritmos de diseño enrevesado. Si un sistema es susceptible de ser evaluado racionalmente, DE puede proporcionar los medios para extraer el mejor rendimiento posible de él.

Las principales características del esquema propuesto en DE, es la utilización del operador genético mutación como un mecanismo de búsqueda y el operador genético selección para dirigir la búsqueda hacia las regiones más prometedoras del espacio de soluciones. Incluye además, dos parámetros propios del algoritmo, el factor de ponderación (F) y el *Crossover Rate* (CR) que influyen en la búsqueda en términos de exploración y explotación respectivamente.

Los Algoritmos Genéticos generan una secuencia de poblaciones usando mecanismos de selección y utilizan el cruce y la mutación como mecanismos de búsqueda. La principal diferencia entre otros algoritmos de la familia de la Computación Evolutiva y la DE, es que otras ramas de la Computación Evolutiva se basan en el cruce, un mecanismo probabilístico de intercambio de información útil entre soluciones para encontrar otras mejores

La idea básica subyacente en DE se traduce en un nuevo esquema para generar vectores de parámetros a los que evaluar. DE genera nuevos vectores de parámetros añadiendo al vector diferencial ponderado—*Weighed Differential Vector*—obtenido a partir de la diferencia y ponderación de dos miembros de población, un tercer miembro o vector de perturbación—*Perturbation Vector*—. Al vector resultante, denominado *Donnor Vector*—vector donante—se le aplica el operador recombinación en conjunto a un vector predeterminado de la población—*Target Vector*—y si el descendiente—*Trial Vector*—produce un valor de la función objetivo inferior a dicho miembro de población predeterminado, este vector recién generado reemplaza al vector predeterminado con el que fue comparado pasando a formar parte de la siguiente generación.

Dicho de otro modo; para cada generación de población de tamaño  $NP$ , será necesario generar una descendencia de  $NP$  *Trial Vectors* que competirán uno a uno con la población actual para determinar la composición de la siguiente generación.

### **El proceso de optimización DE**

En la DE se trabaja con dos matrices de tamaño  $NP \times D$ , donde  $NP$  es el tamaño de la población o cantidad de individuos y  $D$  es el número de dimensiones o parámetros de la función objetivo.

La primera de estas matrices representa la población actual, la otra matriz representa la siguiente generación. Esta última irá almacenando aquellos individuos que superan la fase de selección hasta que finalice la aplicación de operadores biológicos sobre los miembros de la primera matriz, la de la población actual, terminando así una generación.

#### **Inicialización**

En la primera generación, la población se inicializa de forma aleatoria, respetando los intervalos de acotación de cada una de las dimensiones, en caso de no disponer de información sobre el problema. En caso de disponer de información sobre el problema o disponer de una semisolución previa, dicha población puede inicializarse en base a la aplicación de desviaciones normalmente distribuidas respecto a los parámetros de la hipotética semisolución previa.

A cada vector de la población actual, de manera secuencial, se le otorgará el rol de *Target Vector* una única vez por generación. Dicho de otro modo, si recorremos la población, vector a vector, cada índice coincidirá con el *Target Vector* una vez por generación.

#### **Mutación**

Para cada *Target Vector*,  $v_i$ , es preciso generar un *Trial Vector*  $v'_i$  mediante la aplicación de los operadores mutación y recombinación. En el caso básico, la mutación consiste en seleccionar dos miembros de la población aleatoriamente,  $v_{r1}$  y  $v_{r2}$ , asegurándonos de que sus índices no coinciden entre sí ni con el índice del *Target Vector*. La diferencia de estos dos vectores ( $v_{r1} - v_{r2}$ ), produce un nuevo vector, denominado vector diferencial, al que le es aplicado un factor de ponderación  $F$ , constante predefinida perteneciente al intervalo  $(0, 1.0]$ , dando como resultado un vector diferencial ponderado o *Weighed Differential Vector*;  $F(v_{r1} - v_{r2})$  o bien,  $v_w$ .

El vector diferencial ponderado es sumado a otro vector  $v_p$  (ocasionalmente denominado vector de perturbación) extraído también aleatoriamente de la población dando como resultado el vector donante  $v_d$  o *Donnor Vector*:  $v_p + F(v_{r1} - v_{r2})$ . Este vector de perturbación también debe ser distinto a los vectores diferenciales y al *Target Vector* en cuanto a su posición o índice en la población.

### **Recombinación**

A continuación, se aplica el operador recombinación, cruce o *Crossover* entre el *Target Vector* y el *Donnor Vector* en base a un *Crossover Rate* (CR) y un esquema de recombinación para obtener el Vector Trial,  $v'_t$ .

### **Selección**

Por último, se aplica el operador selección entre el *Target Vector* y el *Trial Vector*: aquel cuyo valor de la función objetivo sea menor (para problemas de minimización) pasará a formar parte de la siguiente generación, siendo introducido en la matriz de dicha siguiente generación. Al vencedor se le puede denominar *Survival Vector*,  $v_s$ .

Cada generación termina cuando para cada *Target Vector* ha sido generado un *Trial Vector* e introducido aquel cuya adecuación es superior en la matriz de la descendencia o siguiente generación.

El esquema en sí, presenta elitismo, ya que nunca un *Trial Vector* reemplazará a un *Target Vector* si no tiene un valor de función objetivo menor, hecho que también contribuye a que la DE tienda a la convergencia.

### **Terminación**

El algoritmo DE puede establecer como criterio de finalización:

- Un número determinado de generaciones
- Un tiempo máximo de ejecución
- Encontrar la solución o un candidato a solución con un valor *fitness* esté por debajo de un umbral predeterminado
- Alcanzar un grado predefinido de convergencia (o falta de diversidad) en la población

## **Modelo matemático del Algoritmo *Differential Evolution***

El caso básico de *Differential Evolution* se formula con las siguientes ecuaciones:

### **Obtención del *Donnor Vector* o mutación**

El operador genético de mutación en DE se define mediante la Ecuación 3.1.2-1

$$v_d = v_p + F(v_{r1} - v_{r2})$$

#### **Ecuación 3.1.2-1: *Donnor Vector***

Donde:

- $v_d$  es el *Donnor Vector*
- $v_p$  es el *Perturbation Vector*
- $v_{r1}$  y  $v_{r2}$  son los vectores diferenciales

- $F(v_{r1} - v_{r2})$  se trata del *Weighed Differential Vector*, también denotado como  $v_w$

#### Obtención del *Trial Vector* o recombinación

El operador recombinación o cruce en DE se define mediante la Ecuación 3.1.2-2

$$v'_t(i,j) = \begin{cases} v_{d(i,j)} : rand(i,j) \leq CR \vee j = irand \\ v_t(i,j) : rand(i,j) > CR \wedge j \neq irand \end{cases}$$

#### **Ecuación 3.1.2-2: Trial Vector**

Donde

- $v_t'$  es el Trial Vector
- $v_t$  es el Target Vector
- $rand(i,j)$  es un número aleatorio perteneciente al intervalo [0,1]
- CR es el Crossover Rate
- $irand$  es un número entero aleatorio perteneciente al intervalo [0, D], siendo  $D$  el número de dimensiones de los vectores de la población

#### Obtención del *Survival Vector* o selección

El operador selección en la DE se define mediante la Ecuación 3.1.2-3:

$$v_s = \begin{cases} v'_t : f(v'_t) \leq f(v_t) \\ v_t : f(v'_t) > f(v_t) \end{cases}$$

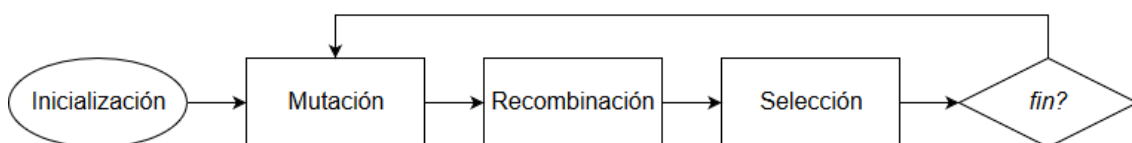
#### **Ecuación 3.1.2-3: Survival Vector**

Donde

- $v_s$  es el vector que formará parte de la siguiente generación
- $v'_t$  es el Trial Vector
- $v_t$  es el Target Vector
- $f$  es la función de evaluación

### **Diagrama del algoritmo *Differential Evolution***

El siguiente diagrama representa de forma esquemática el proceso algorítmico DE explicado en las secciones anteriores, puede expresarse según Figura 3.1.2-1



**Figura 3.1.2-1: Proceso Differential Evolution**

## **Operadores biológicos**

Se revisa el significado de cada uno de los operadores biológicos o genéticos en DE.

### **Operador Mutación**

La aplicación de este operador en DE sirve como motor primario de búsqueda por el espacio de soluciones. En esencia, consiste en obtener un vector expresable como combinación lineal de vectores que existen en la población.

Distintos tipos de mutación se basan en el número de vectores diferenciales que utilizar o si emplear únicamente vectores extraídos aleatoriamente de la población o utilizar otros criterios, como incorporar en la mutación aquél que mejor fitness exhiba en la población actual.

### **Operador Recombinación**

En DE se emplea para dirigir la búsqueda, produciendo un vector cuyos cromosomas—parámetros de la función objetivo—son algunos, los mismos que los de un vector actual de la población y otros tienen valores obtenidos a partir del proceso de mutación.

Fundamentalmente se identifican dos tipos de recombinación, la binaria y la exponencial. Si bien el tipo de operador recombinación explicado en esta sección es de recombinación binaria.

La principal diferencia entre estos dos tipos atiende a razones probabilísticas.

### **Operador Selección**

Es muy simple ya que es un duelo uno a uno, suponiendo un ahorro computacional respecto a otros tipos de selección identificables en distintos Algoritmos Evolutivos. Garantiza que nunca empeore el fitness generación a generación.

## **Parámetros internos del algoritmo**

Por lo general, una combinación de CR y F que favorezca el proceso de optimización depende de diversos factores, por lo que pese a la información que se ofrece a continuación sobre estos parámetros y su configuración, en el caso de enfrentarse a nuevas formas de representación de los individuos, nuevos procesos de decodificación o la utilización de nuevas funciones objetivo, lo recomendable es realizar una exploración de distintas combinaciones de estos parámetros.

### **Factor de ponderación**

El parámetro F, factor de ponderación o escalado tiene un impacto en la amplitud de la búsqueda; cuanto mayor sea el factor de ponderación habrá una mayor exploración por el espacio de soluciones. Para evitar una convergencia prematura, se recomienda que el factor F tenga un valor moderadamente elevado.

En la literatura se mencionan límites superior e inferior para este parámetro. El límite superior teórico se establece en 1.2 de forma empírica: no se han encontrado estudios en



los que para un mayor valor de F se haya resuelto el problema de optimización de forma eficiente. Otros estudios apuntan a que un valor de F comprendido en el intervalo [0.4, 0.9] incrementa la probabilidad de escapar de mínimos locales.

### **Crossover Rate**

El parámetro CR controla cuáles y cuántas variables son mutados en cada individuo. La tasa de cruce CR expresa la probabilidad  $0 \leq CR \leq 1$  de mezclar componentes entre el *Target Vector* y el *Donnor Vector*.

El parámetro CR contribuye en la búsqueda pero en términos fundamentalmente de explotación. En el caso de selección binaria—el mismo que se detalla en esta sección—un mayor valor de CR influye en que habrá más posibilidades de que los cromosomas del *Trial Vector* procedan del *Donnor Vector*, facilitando así la convergencia. Por lo que se deduce que cuanto mayor sea el CR menor será la explotación.

### **Variantes o estrategias**

El caso DE explicado en esta sección corresponde a la estrategia DE/rand/1/bin. En general, las estrategias del algoritmo DE se expresan con el patrón *DE/x/y/z*, donde

- DE son las siglas de *Differential Evolution*
- x es un marco para definir cómo se seleccionan los vectores necesarios para confeccionar el *Donnor Vector*; "rand" significa que se escogerán todos aleatoriamente.
- y es el número de vectores diferenciales que intervendrán en la mutación. Nótese que "1", implica se seleccionarán dos vectores para construir el vector diferencial.
- z hace referencia al tipo de recombinación, que puede ser binaria, "bin", o exponencial, "exp".

Para más información sobre las distintas estrategias consultar (Das, Mullick, & Suganthan, 2016).

### **Un ejemplo ilustrativo**

Dada una población inicializada, ver Figura 3.1.2-2, de NP individuos de D dimensiones cada uno, una función objetivo que minimizar así como un factor de ponderación y un *Crossover Rate* predeterminados:

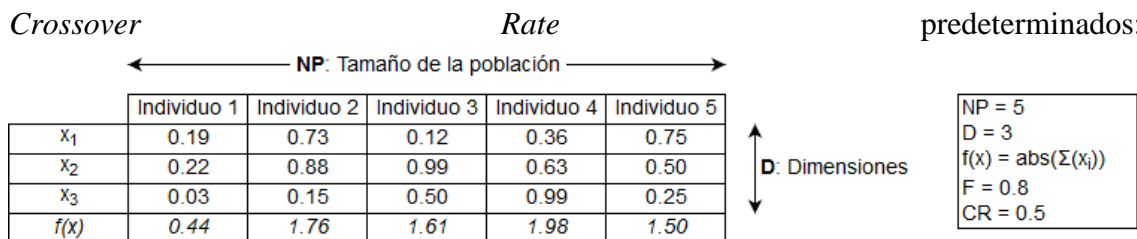


Figura 3.1.2-2: Ejemplo de población inicial en DE

El siguiente paso ilustrado en la Figura 3.1.2-3, consistiría en realizar la mutación, para ello, primero ha de obtenerse el *Weighed Differential Vector*, aplicando el correspondiente factor de ponderación F al vector diferencial.

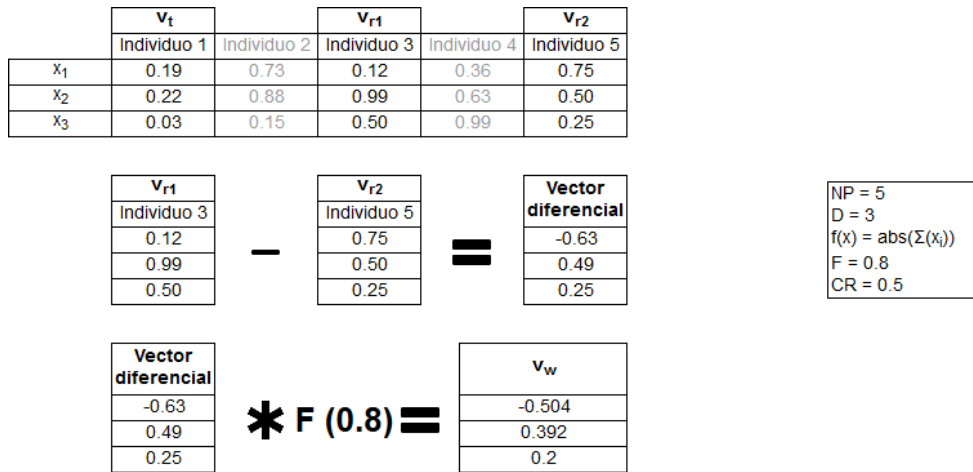


Figura 3.1.2-3: Obtención del Weighed Differential Vector

Lo siguiente, todavía en la etapa de mutación, es obtener el *Donnor Vector*. Para ello se suman el *Perturbation Vector* y *Weighed Differential Vector* como en la Figura 3.1.2-4

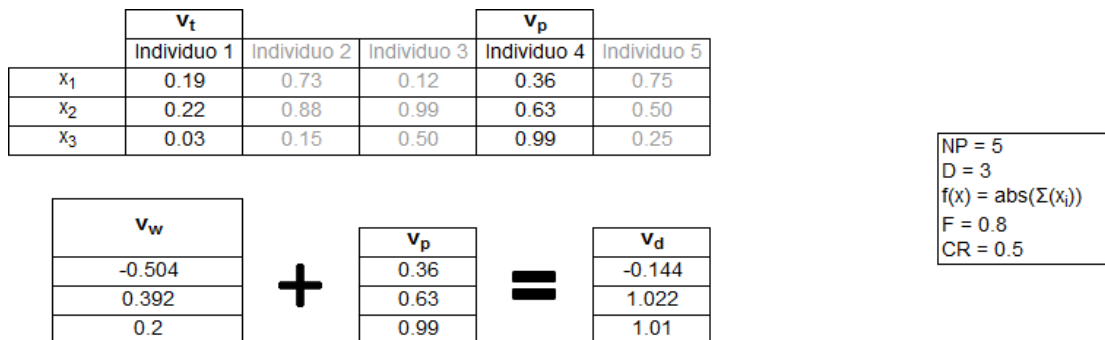


Figura 3.1.2-4: Obtención del Donnor Vector

A continuación se alcanza la etapa de recombinación, donde es obtenido el Trial Vector como descendiente tras aplicar el operador recombinación a los vectores Donnor y Target.

Dicha recombinación se puede observar en la Figura 3.1.2-5.

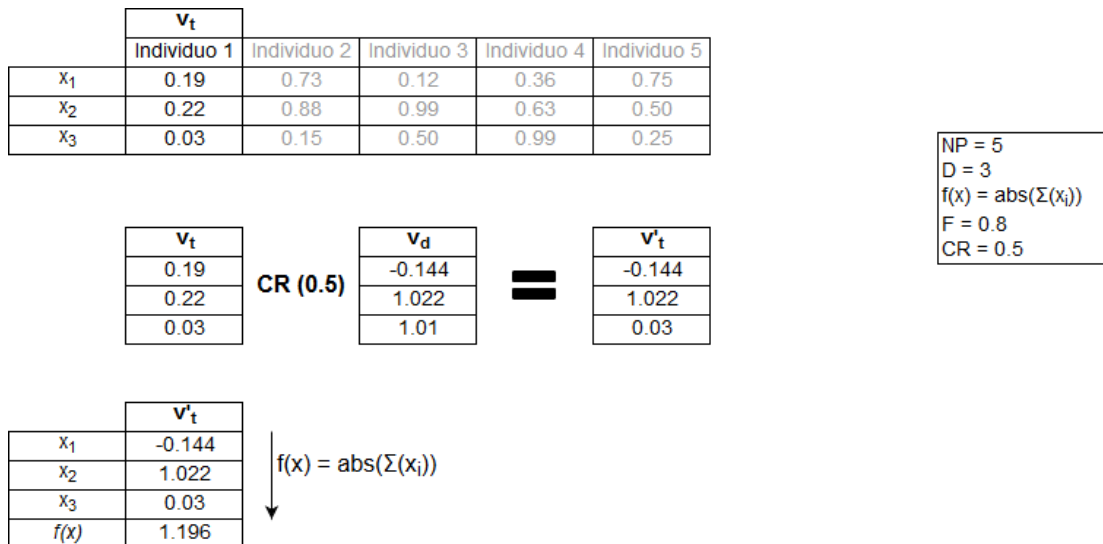


Figura 3.1.2-5: Recombinación en DE

Por último en la etapa de selección, ver Figura 3.1.2-6, se determina el superviviente para la próxima generación eligiendo de entre los vectores Trial y Target aquel cuyo fitness sea menor (para problemas de minimización).

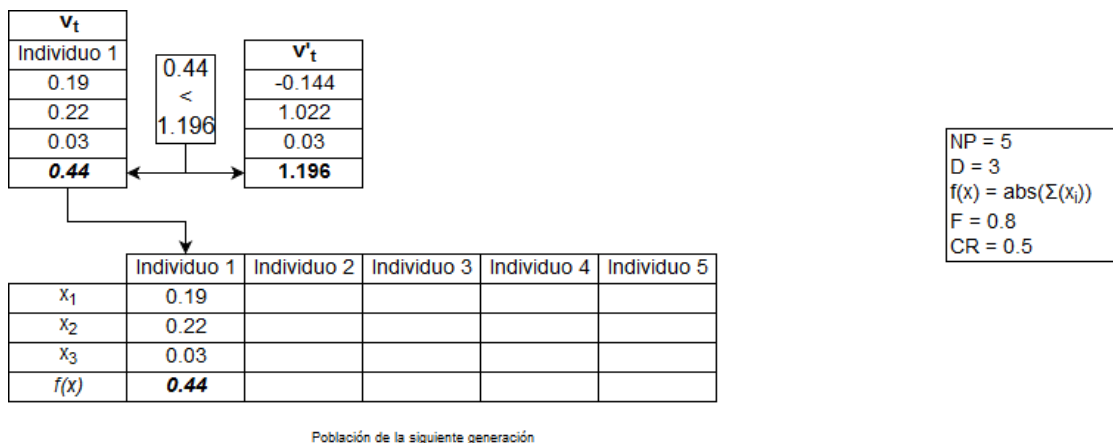


Figura 3.1.2-6: Superviviente para la siguiente generación

## 3.2. Descripción del método

En esencia, el método funciona igual que el ejemplo visto en la sección anterior, pero empleando una función objetivo basada en el la comparación de la distancia Euclídea entre el descriptor LBP de un template y el descriptor LBP de individuo. La complejidad reside fundamentalmente en la decodificación de los individuos.

Dado el proceso descrito y que se está utilizando un EA, es preciso determinar el modo en que se representan los individuos, cómo se decodifican y qué función *fitness* se busca optimizar. En las siguientes subsecciones se describen todas estas cuestiones.

### 3.2.1. Codificación de los individuos

Han sido definidos cuatro modelos de individuo para este estudio, cada uno de ellos orientado a poder resolver cada uno de los casos planteados en la descripción del problema. Cuando hablamos de modelos de individuo nos estamos refiriendo a la representación del genotipo o candidato a solución dentro del espacio de soluciones del problema.

Cada modelo se distingue de los demás por el número de dimensiones y el rol que cumple de cada una de dichas dimensiones en la decodificación, resumido en la Tabla 3.2.1-1.

Nombre del modelo	Vector genotipo	Representación
TLF	$[x, y]$	Esquina superior izquierda desde la que extraer una subimagen de la escena
TLFS	$[x, y, \alpha]$	Factor de escalado $\alpha$ aplicado a la subimagen individuo tras ser extraída de la escena
TLFR	$[x, y, \theta]$	Ángulo de rotación $\theta$ aplicado a la subimagen individuo tras ser extraída de la escena
TLFSR	$[x, y, \alpha, \theta]$	Combina cambio de escala y rotación durante el proceso de decodificación

Tabla 3.2.1-1: Modelos de genotipo

### 3.2.2. Proceso de decodificación de los individuos

A alto nivel, la decodificación sigue el siguiente proceso, de la Figura 3.2.2-1.

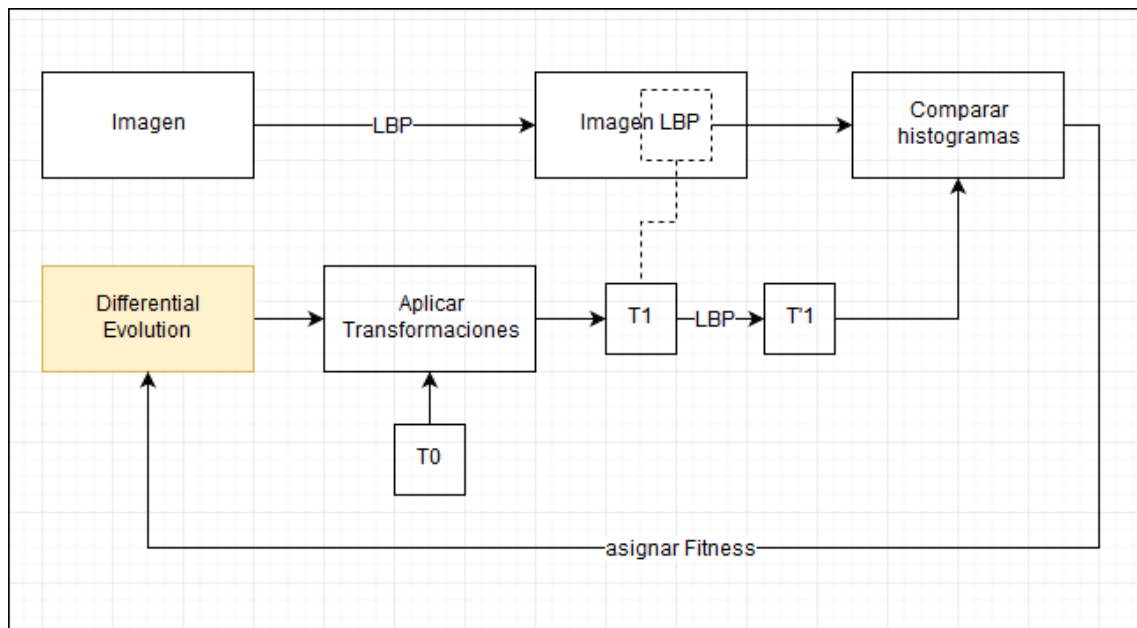


Figura 3.2.2-1: Esquema a alto nivel del método propuesto

Dada una imagen escena y un template (T0), el Algoritmo Evolutivo se basa en las dimensiones del genotipo que representan cambio de escala y rotación para aplicar las transformaciones afines definidas por tales dimensiones al template T0, produciendo un template T1.

En base a las coordenadas x e y, se extrae una subimagen de la escena que concuerde con las dimensiones del template T1.

Tras calcular los histogramas del template T1 y la subimagen extraída de la escena, se le asigna un coste a un individuo.

Dependiendo de las transformaciones que haya que hacer—en base al tipo de problema que se persigue resolver—es preciso tener en cuenta una serie de consideraciones. Para más detalle consultar el Anexo B,

Procesos de decodificación detallados.

### 3.2.3. Función Fitness

#### Función Fitness Básica

La Ecuación 3.2.3-1 describe la función *fitness* básica empleada en este estudio:

$$f = \frac{dist(hist(LBP(t)) - hist(LBP(i)))}{r}$$

Ecuación 3.2.3-1: Función Fitness básica

Donde:

- *t* es la matriz imagen (escala de grises) del template
- *i* es la matriz imagen (escala de grises) del individuo
- *dist* es la distancia Euclídea, entre el vector de características LBP del template y el vector de características LBP del individuo.
- *LBP* es el operador LBP
- *hist* referencia el vector de características resultante de la aplicación del operador LBP
- *r* es el resto, obtenido a través de la Ecuación 3.2.3-2:

$$r = \Delta x * \Delta y - (2 * \Delta x + 2 * (\Delta y - 2))$$

Ecuación 3.2.3-2: Resto (píxeles que sobran)

Donde:

- $\Delta x$  es la anchura del template (y el individuo)
- $\Delta y$  es la altura del template (y el individuo)

## **CAPÍTULO 4. RESULTADOS EXPERIMENTALES Y DISCUSIÓN**

---

## 4.1. Consideraciones previas

Para la correcta exposición de los resultados experimentales y para facilitar su interpretación, se procede a realizar una breve descripción de los valores estadísticos analizados así como de los campos de las tablas incluidas en el Anexo A.

### 4.1.1. Estadísticos propios del Algoritmo Evolutivo

La validación del método se enfoca en determinados criterios, algunos de ellos son estadísticos habituales de la Computación Evolutiva

#### ***Success Rate (SR)***

Se trata de un criterio para determinar si un run de una ejecución puede considerarse un éxito, es decir, se ha encontrado la solución o se ha superado un umbral predefinido que permite considerar un candidato a solución como tal.

Para todos los experimentos se ha utilizado el mismo criterio. Conocido el centro o coordenadas de Ground Truth de la solución y las coordenadas de extracción del template, se considera que se ha alcanzado el éxito en el run si la distancia entre las coordenadas de ambos centros es inferior a una distancia prefijada, como indica Ecuación 4.1.1-1:

$$s_i = \begin{cases} 1 & \leftrightarrow \text{dist}(c, c') \leq d \\ 0 & \leftrightarrow \text{dist}(c, c') > d \end{cases}$$

**Ecuación 4.1.1-1: Umbral para el criterio de éxito**

Donde:

- $s_i$  es la determinación de éxito del run de índice  $i$  de una ejecución (1 es un éxito, 0 es un fracaso)
- $\text{dist}$  es la distancia euclídea
- $c$  son las coordenadas del Ground Truth o centro de la subimagen template
- $c'$  es son las coordenadas de la subimagen individuo
- $d$  es el umbral predefinido

Finalmente, la formulación matemática del SR para estos experimentos viene dada por la Ecuación 4.1.1-2:

$$SR = \frac{\sum_{i=1}^n s_i}{n}$$

**Ecuación 4.1.1-2: Tasa de Éxito (SR)**

Donde:



- $i$  es el valor de éxito del run de índice  $i$
- $n$  es la cantidad total de runs de la ejecución

### **Mean Best Fitness (MBF)**

El *Mean Best Fitness* es la el promedio de los mejores fitness encontrados en cada run de una ejecución. Sigue la siguiente formulación, según la Ecuación 4.1.1-3:

$$MBF = \frac{\sum_{i=1}^R bf(r)}{R}$$

**Ecuación 4.1.1-3: Mean Best Fitness**

Dónde:

- $R$  es el número de runs
- $i$  es el índice del run actual
- $bf(r)$  es el valor del mejor fitness encontrado en el run

### ***Average Evaluations to Solution (AES)***

Se trata del promedio del número de veces que ha sido invocada la función *fitness* hasta que es encontrado un individuo que cumple el criterio de éxito, de entre aquellos runs donde se haya alcanzado el éxito.

### **4.1.2. Descripción de los campos de las tablas de resultados**

Todas las tablas cuyos campos se describen en esta sección se mostrarán todas ellas en el Anéxo A:, ubicación donde han sido trasladadas las tablas con el objeto de centralizar los resultados obtenidos en todos experimentos y facilitar el análisis de los mismos.

Cada tabla de resultados contiene una cabecera donde se especifican los distintos aspectos analizados en el experimento. Cada fila corresponde a una ejecución, una ejecución es un conjunto de uno o más runs donde los parámetros tanto del algoritmo evolutivo como del operador LBP han sido fijados. Es especialmente importante esta consideración para entender los resultados y las gráficas comparativas en todos los experimentos.

A continuación se listan los campos o columnas donde se documentan los resultados de los distintos experimentos y se indican una serie consideraciones con el fin de facilitar la

- Campo "F": Indica el valor del parámetro F o factor de ponderación del algoritmo evolutivo
- Campo "CR": Indica el valor del parámetro CR, Crossover Rate del algoritmo evolutivo
- Campo "D": Enfatiza el número de dimensiones del genotipo

- Campo "NP": Indica el tamaño de la población empleado para los runs de la ejecución
- Campo "R": Se trata del número de runs (solo para experimento 2)
- Campo "GR": Indica el número de generaciones del run corresponde también en este caso al criterio de finalización del run
- Campo "SRT": Se trata de una distancia mínima, umbral, que debe darse entre las coordenadas del centro de una subimagen individuo respecto a las coordenadas del template o Ground Truth de la escena para considerar que se ha alcanzado un éxito en el run.
- Campo "SR": Se trata de la tasa de éxito
- Campo "ICD": Es la distancia intercentral de las coordenadas del template o Ground Truth respecto a las coordenadas del centro del individuo cuyo fitness ha sido el más bajo en toda la ejecución. Nótese que no tiene por qué coincidir que el mejor fitness encontrado en la ejecución se considere un éxito.
- Campo "MBF  $\pm$  STD": Se trata del Mean Best Fitness y su desviación estándar
- Campo "AES": Se trata del Average Evaluations to Solution obtenido para la ejecución
- Campo "TOTAL TIME": Duración total de la ejecución
- Campo "TPR AVG  $\pm$  STD": Duración media del conjunto de runs de la ejecución junto a su desviación estándar
- Campo "Best fitness": Se trata del menor fitness obtenido para esta ejecución. Nótese que no tiene por qué coincidir que el mejor fitness encontrado en la ejecución se considere un éxito.

## 4.2. Experimento 1: Exploración del caso base

### 4.2.1. Objetivo

Existen dos objetivos fundamentales para este primer experimento:

1. Estudiar el rendimiento de la metodología según se añade complejidad a la representación del individuo y por tanto la decodificación del mismo
2. Realizar una primera exploración relativa a qué combinación de parámetros del algoritmo DE pudiera ofrecer mejor rendimiento en términos de SR y MBF.

### 4.2.2. Configuración del experimento

En este experimento se busca verificar la validez de la metodología según es incrementada la complejidad de la representación del individuo y el modelo de decodificación, mientras es mantenida la misma función de evaluación para todas las ejecuciones del experimento.

Al mismo tiempo, se persigue encontrar una combinación de parámetros internos del algoritmo DE (F y CR) que ofrezca buenos resultados en todos los casos.

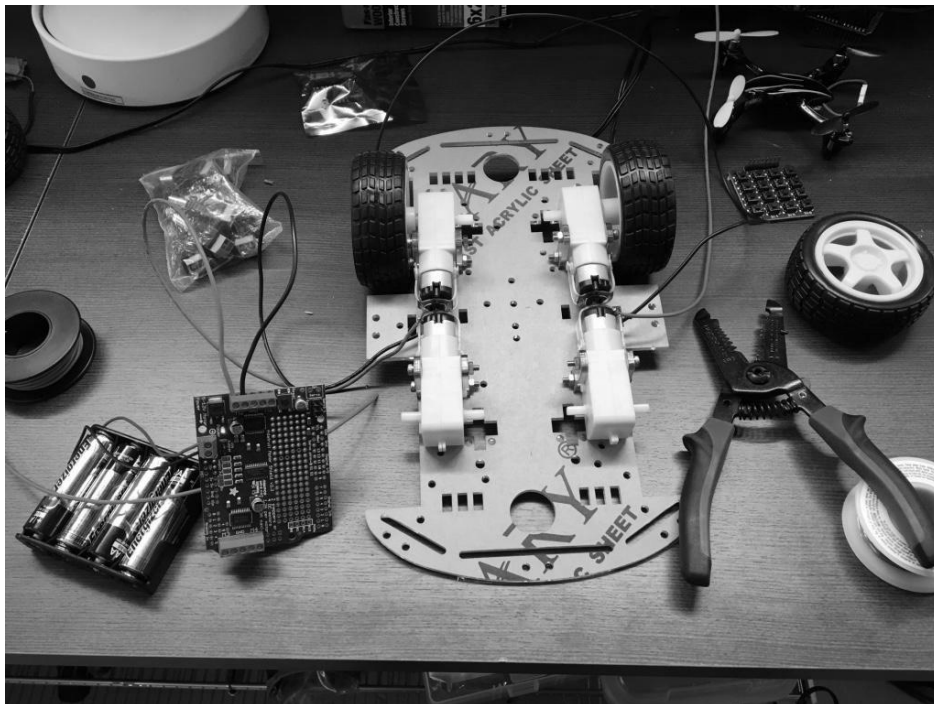
De manera que, para cada uno de los cuatro tipos de representación de individuo y decodificación, se realizan nueve ejecuciones correspondientes a combinar los valores del conjunto {0.2, 0.5, 0.8} para F y CR respectivamente.

Adicionalmente, según es incrementada la complejidad de la representación, para cada dimensión adicional añadida al genotipo, se amplía el número de generaciones en cien unidades.

En total para este experimento han sido realizadas 36 ejecuciones.

### **Imagen escena**

La siguiente imagen, Ilustración 4.2.2-1, es la escena original del experimento. Para el experimento, la imagen es convertida de RGB a escala de grises.



**Ilustración 4.2.2-1: Escena experimento exploración del caso base**

### **Templates**

Los templates elegidos para este experimento son cuatro, cada uno de ellos es destinado a un modelo de individuo y decodificación. Aunque similares, existen diferencias entre los mismos en cuanto al tamaño y la rotación.

El template para el caso “TLF” no es más que una subimagen de la escena. Para el caso “TLFS” se emplea el mismo template que para el caso “TLF”, pero aplicando a éste un cambio de escala con factor 0.7.

El template “TLFR” consiste en un template extraído de la escena tras haber sido esta escena rotada 45° en el sentido contrario a las agujas del reloj. Finalmente, para el caso más complicado donde se incluyen cambio de escala y rotación, “TLFSR”, se ha seguido la misma mecánica y sencillamente se aplicado un cambio de escala factor 0.7 al template que ya incluía la rotación. Se resume en la Tabla 4.2.2-1

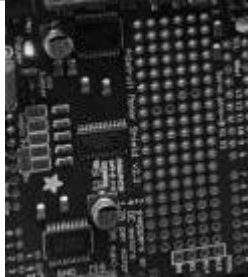

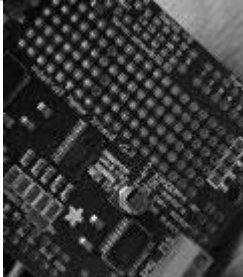
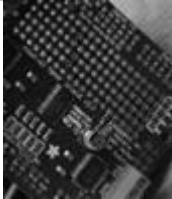
TLF: [x, y]	TLFS: [x, y, $\alpha$ ]	TLFR: [x, y, $\theta$ ]	TLFSR: [x, y, $\alpha$ , $\theta$ ]
			

Tabla 4.2.2-1: Templates experimento 1

### 4.2.3. Análisis de los resultados

Primero se analizarán los resultados de cada modelo de individuo por separado, y a continuación se contrastarán los resultados globales con los objetivos del experimento.

Atendiendo a la Tabla A.1-1: Resultados decodificación básica TLF, observamos que el modelo de individuo y decodificación más simple, TLF, encuentra individuos lo suficientemente cercanos a la solución para cualquier combinación de F y CR, logrando así un SR del 100% en todas las ejecuciones.

Concretamente, los centros de los mejores individuos encontrados (cada mejor individuo de cada una de las ejecuciones) están a menos de 7 píxeles del centro del template original en la escena. Cabe destacar el hecho de que la ejecución donde la ICD es menor, no resulta ser aquella donde el Best Fitness es menor. Esto significa que hay zonas muy cercanas a la solución que tienen un fitness un tanto peor que otras zonas un poco más alejadas.

El MBF se sitúa en el orden de magnitud de las milésimas, y la desviación estándar del MBF llega a ser 0 en dos ejecuciones. También cabe destacar que la duración de cada ejecución es inferior al minuto, rondando un de promedio 45 segundos, por lo que un run apenas dura 5 segundos de media.

Atendiendo a los gráficos de evolución del mejor fitness de los runs de cada ejecución, se puede apreciar que en realidad el límite de generaciones podría haberse reducido a 25 sin que se hubiera perdido demasiada precisión en los resultados.

En los resultados para el modelo TLFS en la Tabla A.1-2: Resultados decodificación básica TLFS, el SR también ha sido del 100% en todas las ejecuciones, pero la ampliación del número de generaciones a 150 y la población a 30 individuos, implica que el tiempo de ejecución se triplique con respecto al modelo básico.

La ICD se mantiene por debajo de los 7 píxeles, incluso en tres ejecuciones dicha distancia apenas en un píxel y medio. En este caso la ejecución donde se encuentra la solución con el menor ICD en esta ocasión esta ejecución sí resulta ser aquella donde se ha encontrado el mejor fitness.

Los valores del MBF se mantienen en el mismo orden de magnitud que para el caso básico, aunque puede apreciarse un ligero incremento en estos valores, así como en la desviación estándar del MBF.

El caso donde se contempla rotación, Tabla A.1-3: Resultados decodificación básica TLFR, teniendo el mismo número de generaciones, 150, y población, 30, que el caso donde se contempla el cambio de escala, debido a que el proceso de decodificación es más elaborado, la duración media de cada run se vuelve a incrementar, cada run dura unas cinco veces más que el run del caso básico, y unos dos tercios más de tiempo que en el caso del cambio de escala, llegando a los 26 segundos de media cada run.

Sin embargo, el SR se sigue preservando al 100%: la ICD se mantiene inferior a 7 píxeles, obteniendo incluso valores por debajo de los 2 píxeles de distancia en dos ejecuciones.

El MBF continúa estable en el orden de magnitud de las milésimas, con una desviación estándar poco elevada.

Alcanzado el caso más complejo, donde se contemplan tanto cambio de escala como rotación, Tabla A.1-4: Resultados decodificación básica TLFSR, a priori se incrementa el tamaño de la población inicial a cuarenta y se conceden 100 generaciones más por run llegando a las 250. El tiempo de ejecución vuelve a incrementarse, el tiempo medio de cada run llega a los 45 segundos, suponiendo un incremento de nueve veces más que el caso básico.

Se vuelve a lograr de este modo un SR del 100%, donde la ICD solo supera levemente los 7 píxeles en tres ejecuciones. Por el contrario, en otras tres ejecuciones la ICD es inferior a dos píxeles. De nuevo, se vuelve a apreciar que existe una ejecución donde el ICD es 5.5334 píxeles y sin embargo en esta ejecución se ha encontrado el mejor fitness, a pesar de como ya ha sido mencionado, existir tres ejecuciones en donde los centros de los mejores individuos encontrados están a menor distancia del centro del template original.

Atendiendo a los mejores genotipos encontrados, podemos observar que las dimensiones del cambio de escala y la rotación han sido aproximadas a la solución con una precisión, que se aleja como mucho 2 centésimas para el cambio de escala y un grado para la rotación.

En términos generales, según las valoraciones realizadas previamente, puede deducirse que para garantizar el éxito a la hora de hallar una solución aceptable solventando al mismo tiempo la dificultad que implica la decodificación de los individuos cuando se añaden dimensiones adicionales, la estrategia planteada, ha sido suficiente. Dicha estrategia ha consistido únicamente establecer la población a 10 individuos por dimensión, y ampliar el número de generaciones en 100 por cada dimensión agregada.

En relación a la combinación de parámetros que mejores resultados han ofrecido, dado que el SR siempre ha sido del 100% en todas las ejecuciones, y que el MBF ha sido bastante estable en todos los casos, centramos la atención a la ICD y al *Best Fitness*, teniendo en cuenta que se han dado situaciones en las que una menor ICD no ha correspondido necesariamente al mejor *Best Fitness*. En base a estas consideraciones, los hallazgos de las mejores combinaciones son ilustrados en la Tabla 4.2.3-1:

Caso	Mejor combinación por fitness	Mejor combinación por ICD
<b>TLF</b>	F = 0.2, CR = 0.2 F = 0.2, CR = 0.5 F = 0.5, CR = 0.2 F = 0.5, CR = 0.5 F = 0.8, CR = 0.2 F = 0.8, CR = 0.5	F = 0.8, CR = 0.8
<b>TLFS</b>	F = 0.2, CR = 0.2	F = 0.2, CR = 0.2
<b>TLFR</b>	F = 0.2, CR = 0.2 F = 0.2, CR = 0.5	F = 0.2, CR = 0.8
<b>TLFSR</b>	F = 0.2, CR = 0.2	F = 0.2, CR = 0.8

Tabla 4.2.3-1: Resumen de las mejores combinaciones F y CR

Esto supone un dilema, ya que salvo para el caso TLFS, no se ha encontrado una combinación de los parámetros internos de DE que satisfaga encontrar la solución más cercana en términos espaciales y que esta al mismo tiempo tenga el mejor fitness.

Por otra parte, haber trabajado con un template similar para cada modelo de individuo, haber encontrado una combinación que cumpliera las premisas deseadas tampoco garantizaría que dicha combinación fuese universalmente adecuada para cualquier tipo de template.

Esta situación da lugar a continuar buscando una buena combinación de parámetros teniendo en cuenta hipotéticas diferencias en la distribución del histograma LBP de otros posibles templates, centrándonos en el modelo de individuo que mayor interés ofrece, TLFSR.

## **4.3. Experimento 2: Exploración de diferentes tipos de templates**

### **4.3.1. Objetivo**

Los objetivos esenciales de este experimento son:

1. Determinar qué combinación de parámetros internos del algoritmo DE ofrecen buenos resultados para distintos templates en el caso TLFSR
2. Analizar el hipotético impacto que pudiera suponer la utilización de templates homogéneos (en cuanto a sus componentes LBP) sobre la metodología planteada

### **4.3.2. Configuración del Experimento**

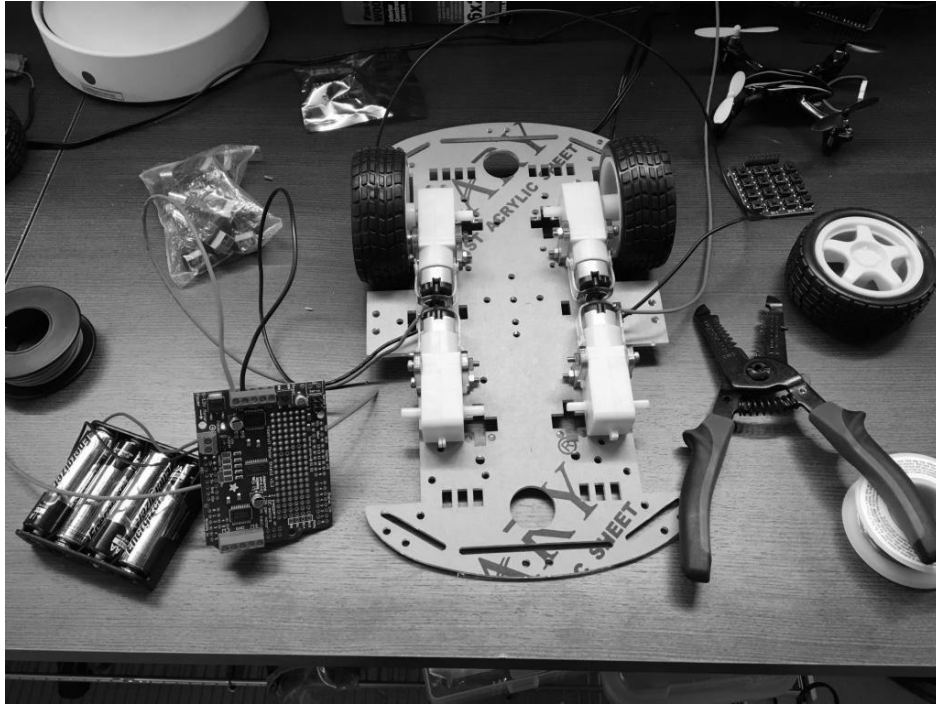
Para este experimento se realizan 36 ejecuciones distribuidas del siguiente modo

- 9 ejecuciones para el template "pilas" probando las combinaciones de F y CR posibles del intervalo {0.2, 0.5, 0.8} limitando el número de generaciones a 250
- 9 ejecuciones para el template "rueda" probando las combinaciones de F y CR posibles del intervalo {0.2, 0.5, 0.8} limitando el número de generaciones a 250
- 9 ejecuciones para el template "pilas" probando las combinaciones de F y CR posibles del intervalo {0.2, 0.5, 0.8} limitando el número de generaciones a 350
- 9 ejecuciones para el template "rueda" probando las combinaciones de F y CR posibles del intervalo {0.2, 0.5, 0.8} limitando el número de generaciones a 350

Preliminarmente, se baraja la hipótesis de que para estos templates sea necesario aumentar el número de generaciones para alcanzar mejores tasas de éxito. Esta es la razón por la que primero se fija el número de generaciones a 250 como en el experimento anterior y posteriormente se amplían a 350 para verificar que la menor complejidad del histograma LBP de estos templates pudiera requerir esta ampliación del número de generaciones.

### **Imagen escena**

La siguiente imagen es la escena original del experimento. Para el experimento, la imagen es convertida de RGB a escala de grises. Ver Ilustración 4.3.2-1:



**Ilustración 4.3.2-1: Escena experimento 2**

## **Templates**

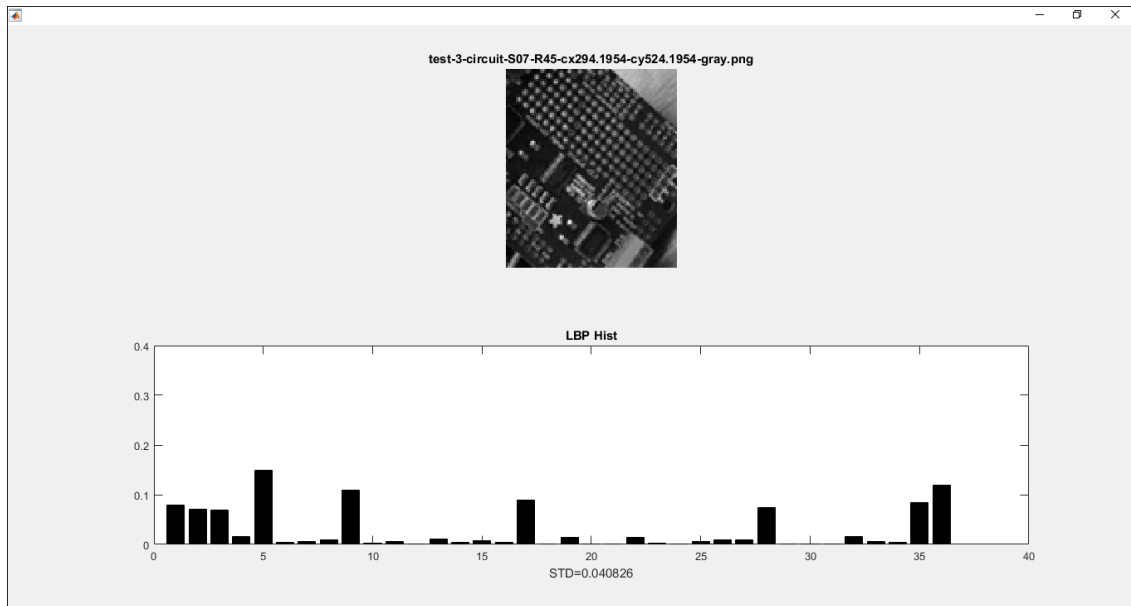
Todos los templates de este experimento se corresponden al caso TLFSR a los cuales les han sido aplicadas las mismas transformaciones geométricas tanto para el cambio de escala (0.7) como para la rotación ( $45^\circ$ ).

A continuación se presentan los templates empleados junto a sus histogramas LBP normalizados así como la desviación estándar o típica de cada uno de ellos.

### **Template circuito**

Se trata del mismo template que el que fuera utilizado para el experimento "Exploración paramétrica básica", aunque el nombre del archivo de la imagen no sea el mismo. Dado que ya se realizaron 9 ejecuciones para este mismo template en el experimento anterior, en esta ocasión no se ha procedido a realizar ninguna ejecución adicional. Únicamente es plasmado el histograma LBP normalizado con el fin de poder realizar comparaciones con los otros dos templates sobre los que sí se realizan ejecuciones en este experimento.





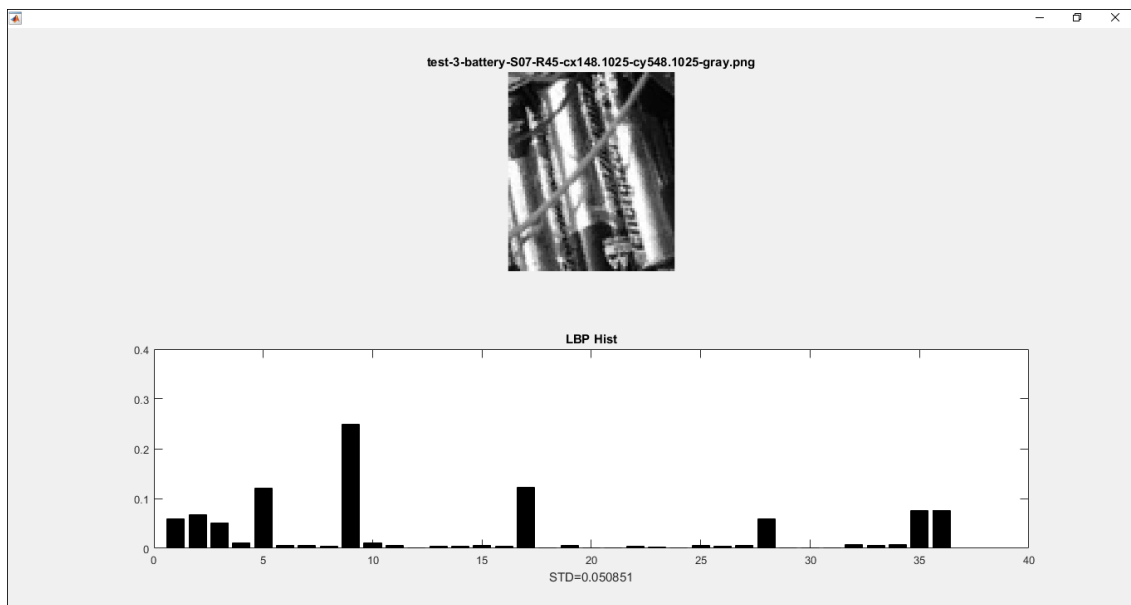
**Ilustración 4.3.2-2: Template circuito e histograma LBPri en experimento 2**

Podemos observar en la Ilustración 4.3.2-2, mediante el histograma LBP de la imagen del circuito, que se aprecia la aparición de numerosos componentes LBP, debido a las características accidentadas de la textura del template.

La desviación estándar, se sitúa en 0.040826. Asumimos la noción de que una desviación estándar pequeña corresponde a histogramas donde hay menos varianza entre los distintos componentes LBP, por lo que la textura de la imagen es más "compleja".

### Template pilas

Se trata de una subimagen escalada y rotada donde aparecen unas pilas y un cable.

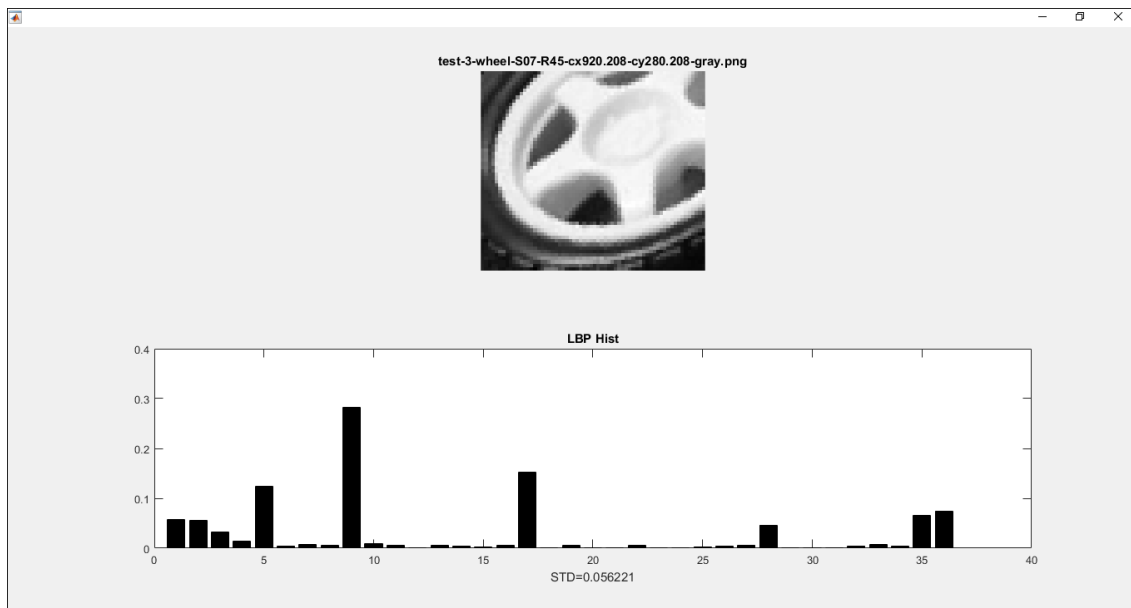


**Ilustración 4.3.2-3: Template pilas e histograma LBPri del experimento 2**

En la Ilustración 4.3.2-3, pueden apreciarse más zonas con textura uniforme que en el caso del circuito. Observamos que la desviación estándar aumenta a 0.050851, y se verifica en el histograma LBP la aparición de numerosos componentes con pocas ocurrencias.

### **Template rueda**

Se trata de una subimagen de la escena donde se puede apreciar la llanta y parte del neumático correspondientes a una rueda. El template también ha sido escalado y rotado.



**Ilustración 4.3.2-4: Template rueda e histograma LBPri del experimento 2**

Por último, en el histograma LBP asociado a la imagen de la rueda, Ilustración 4.3.2-4, donde práctica y visualmente la uniformidad de la textura de la imagen es visiblemente mayor, la desviación estándar vuelve a aumentar hasta el valor 0.056221.

En base a este análisis podemos referirnos a la noción intuitiva de que aquellas imágenes con más uniformidad, o aquellas cuya textura sea menos "heterogénea" producirán histogramas LBP con más varianza.

A priori, sin considerar el resto de composición de la escena, puede sospecharse que será más difícil localizar templates cuya uniformidad sea mayor; correlacionado con el hecho de que sus histogramas LBP exhibirán una mayor desviación típica.

### **4.3.3. Análisis de los resultados**

El estudio de los resultados del experimento se plasma del siguiente modo. Primero se analizan los resultados por separado de cada uno de los templates. A continuación se mencionan las posibles causas de los resultados obtenidos a nivel global en relación a la

complejidad de los histogramas. Por último se constatan las combinaciones de los parámetros internos del algoritmo que mejor han funcionado en términos generales.

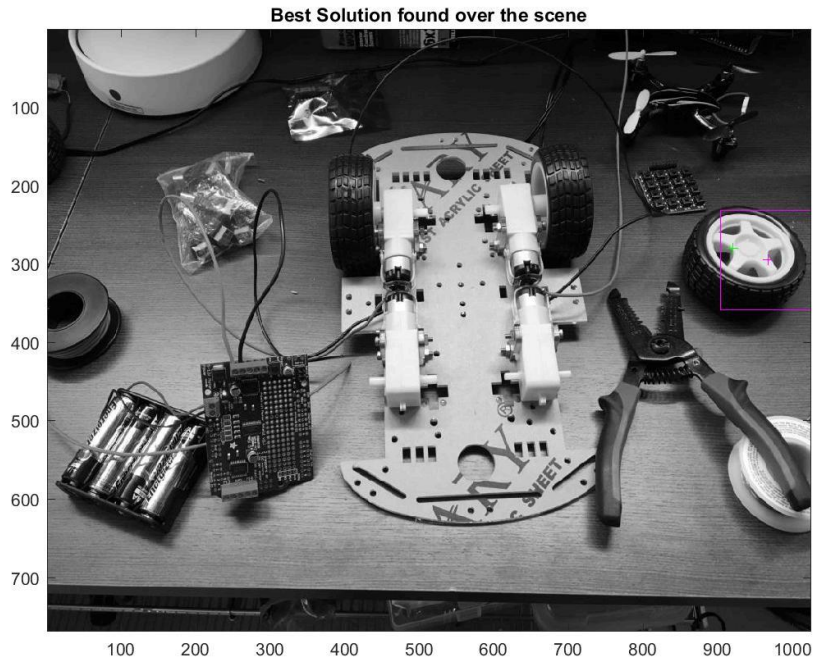
Las primeras 9 ejecuciones del template "pilas" cuyos resultados están disponibles en la Tabla A.2-1: Resultados exploración templates homogéneos, pilas GR=250, arrojan resultados muy irregulares. Podemos observar que la tasa de éxito, SR, solo llega al 100% para dos combinaciones de F y CR, donde F= 0.8 en ambos casos y CR es 0.2 y 0.5. Las distancias intercentrales, ICD, son menores a 7 píxeles salvo en una ejecución. Incluso hay dos ejecuciones donde la distancia intercentral es menor a un pixel, las dos primeras, pero por el contrario la tasa de éxito en estos dos casos es baja, del 40% y 60% respectivamente.

El MBF aumenta un orden de magnitud respecto a la primera batería de experimentos, situándose en el orden de las centésimas en lugar de las milésimas. La desviación estándar sigue siendo bastante estable. El tiempo de ejecución disminuye en relación los casos TLFSR del template circuito, situándose en los 37 segundos de media por run, debido fundamentalmente a que el template de las pilas tiene un tamaño menor que el del circuito.

Los mejores fitness oscilan entre 0.008 y 0.02, detectándose de este modo una cierta variabilidad no hallada anteriormente. Atendiendo a los mejores genotipos hallados, se observa que la distancia de las dimensiones de cambio de escala y rotación son ligeramente superiores también para el caso de este template con el número de generaciones fijada a 250.

Para las primeras 9 ejecuciones del template "rueda", Tabla A.2-2: Resultados exploración templates homogéneos, rueda GR=250, el SR empeora considerablemente, no hay ninguna ejecución donde la tasa de éxito llegue al 100%. Parece no obstante existir una cierta correlación entre los SR encontrados en cada ejecución de este caso con el caso anterior de las pilas; obteniendo la combinación F= 0.8 y CR = 0.2 | 0.5 un SR superior al resto de los casos; 0.7 y 0.6 respectivamente.

Las distancias intercentrales del template original respecto a los mejores individuos encontrados se sitúan mayoritariamente por debajo de los 7 píxeles o cercanos a 7 salvo en un caso particular donde dicha distancia se sitúa por encima de los 50 píxeles y por tanto ha de entenderse que el mejor individuo encontrado en esa ejecución respecto al fitness, ni siquiera se corresponde con un caso considerado un éxito. Se trataría pues de un óptimo local. La siguiente Ilustración 4.3.3-1, corresponde a dicho caso.



**Ilustración 4.3.3-1: Óptimo local hallado en experimento 2**

El MBF vuelve a situarse en el orden de magnitud de las centésimas, con una desviación también poco elevada. El tiempo de ejecución por run vuelve a disminuir, hasta los 31 segundos de media, lo cual no es ninguna sorpresa dado que este template tiene un tamaño todavía menor al template "pilas".

Los valores de mejores fitness quedan situados entre los valores 0.011 y 0.026, estrechándose el intervalo de los mismos. En cuanto a los mejores genotipos, y prestando especial atención a las dimensiones del cambio de escala y la rotación, se observa que los intervalos aumentan en ambos casos respecto a experimentos anteriores siendo para el cambio de escala [1.22, 1.48] y para la rotación [37°, 47°].

Ampliando el número de ejecuciones a 350 para el template "pilas", Tabla A.2-3: Resultados exploración templates homogéneos, pilas GR=350, el panorama para el SR se mantiene o mejora en casi todos los casos pero no en todos, destacando eso sí que las combinaciones de parámetros internos de DE que alcanzaron un SR del 100% se mantienen.

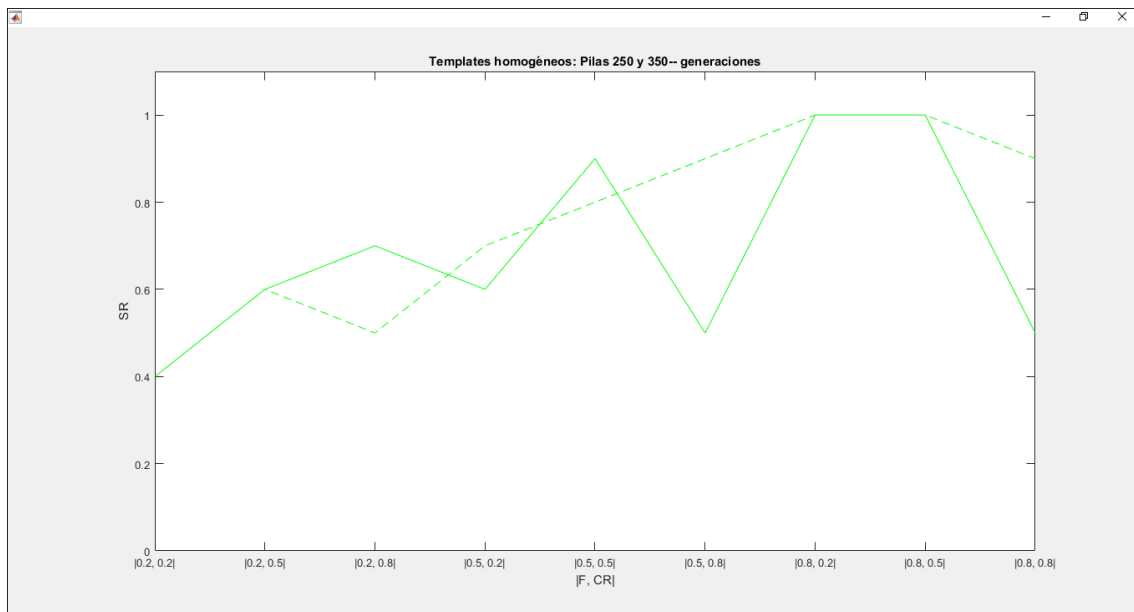
Estas 100 generaciones adicionales suponen un incremento medio de tiempo por run de unos 22 segundos para este template. Se aprecia una mejoría en las distancias intercentrales obtenidas, encontrando un único caso que supera los 7 píxeles de distancia.

Los mejores fitness tienden a bajar ligeramente en la mayoría de las ejecuciones. Por otra parte, los mejores genotipos estrechan el cerco a la solución en las dimensiones del cambio de escala y la rotación.

Por último, la ampliación de 100 generaciones para el caso del template "rueda", Tabla A.2-4: Resultados exploración templates homogéneos, rueda GR=350, no supone una mejora de la tasa de éxito en términos generales, salvo para las combinaciones  $F = 0.8$  y  $CR = 0.2 \mid 0.5$ , donde se alcanza el 90%, lo cual es un dato muy sugerente de cuál podría ser una buena combinación de parámetros estándar para el problema planteado.

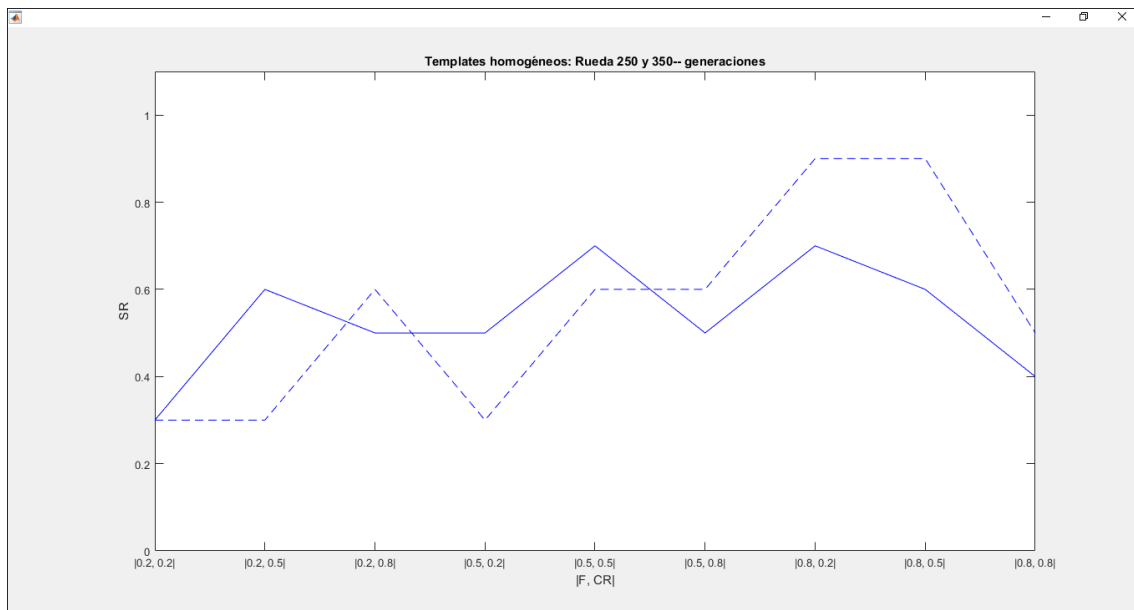
Las ICD encontradas mejoran notablemente tras ampliar el número de generaciones, quedando todas ellas por debajo de los 7 píxeles salvo una ejecución donde dicha distancia son precisamente 7 píxeles. El MBF se mantiene en el mismo orden de magnitud, pero se aprecia un decremento del mismo en la mayoría de las ejecuciones. La desviación estándar ofrece resultados muy similares a las ejecuciones limitadas a 250 generaciones.

El tiempo de ejecución por run asciende 15 segundos de media, alcanzando para cada run los 46 segundos de media. Se aprecia una discreta mejora general en los mejores fitness encontrados. Por otra parte, en los mejores genotipos es especialmente llamativo el hecho de que ningún valor del cambio de escala en estos genotipos alcance el valor 1.4. El valor de la dimensión de la rotación en cambio varía de los  $42^\circ$  a los  $49^\circ$ .



**Figura 4.3.3-1: Comparación template pilas 250GR vs 350GR experimento 2**

Puede apreciarse atendiendo a Figura 4.3.3-1, que la tasa de éxito para el template pilas alcanza el 100% en las mismas combinaciones de  $F$  y  $CR$ ,  $\{0.8, 0.2\}$  y  $\{0.8, 0.5\}$  para las ejecuciones de 250 y 350 generaciones.



**Figura 4.3.3-2: Comparación template rueda 250GR vs 350GR experimento 2**

En la Figura 4.3.3-2, otra puede apreciarse una cierta tendencia en la mejora en la mejora del SR para valores altos de F y un mayor número de generaciones.

En términos generales, tras la realización de este experimento todo parece indicar que las mejores combinaciones de parámetros obtenidas son como ya ha sido mencionado anteriormente  $F=0.8$  y  $CR=0.2$  o  $CR=0.5$  en cuanto a las tasas de éxito obtenidas para estas dos combinaciones sobre estos dos modelos de template.

Es preciso señalar, que la homogeneidad de un hipotético template es un factor especialmente relevante a tener en consideración, ya que la influencia de dicha homogeneidad es crucial para el criterio de tasa de éxito empleado.

La heterogeneidad inherente de los *templates* anima a comprobar si una exploración más distante del vecindario en el operador LBP o la utilización de una función de coste que contemplase los tres canales RGB podría solventar la dificultad de determinados tipos de template.

En cambio, las observaciones realizadas sobre los mejores *fitness* encontrados, sugieren introducir una etapa de búsqueda local para terminar de dar con la solución global.

## 4.4. Experimento 3: Influencia de la variación de intensidad luminosa

La variación de la intensidad luminosa consiste en mapear los valores de los píxeles de una imagen I a nuevos valores en una imagen J de acuerdo a dos intervalos definidos, uno para cada mapa.

En este caso, el intervalo de entrada que se basa las intensidades mínima y máxima de la imagen original, y el intervalo de salida es  $[0, 1]$ .

Un parámetro  $\gamma$  interviene en el mapeo para definir la forma de la curva que describirá las relaciones entre los píxeles de las imágenes I y J

#### **4.4.1. Objetivo**

Los objetivos esenciales de este experimento son:

1. Comprobar la tolerancia del operador LBP a cambios de intensidad luminosa constante.
2. Comprobar la influencia de los cambios de intensidad luminosa constante sobre las distintas combinaciones de los parámetros DE.

#### **4.4.2. Configuración del Experimento**

El experimento consta de 81 ejecuciones según la siguiente organización:

Aplicación de una variación de intensidad luminosa con  $\gamma = 2.0$ , una saturación de entrada y una de salida en el intervalo  $[0, 1]$  sobre la imagen escena

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto  $\{0.2, 0.5, 0.8\}$
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto  $\{0.2, 0.5, 0.8\}$
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto  $\{0.2, 0.5, 0.8\}$

Aplicación de una variación de intensidad luminosa con  $\gamma = 1.0$ , una saturación de entrada y una de salida en el intervalo  $[0, 1]$  un sobre la imagen escena

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto  $\{0.2, 0.5, 0.8\}$
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto  $\{0.2, 0.5, 0.8\}$
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto  $\{0.2, 0.5, 0.8\}$

Aplicación de una variación de intensidad luminosa con  $\gamma = 0.5$ , una umbralización de entrada y una de salida en el intervalo  $[0, 1]$  un sobre la imagen escena

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto  $\{0.2, 0.5, 0.8\}$

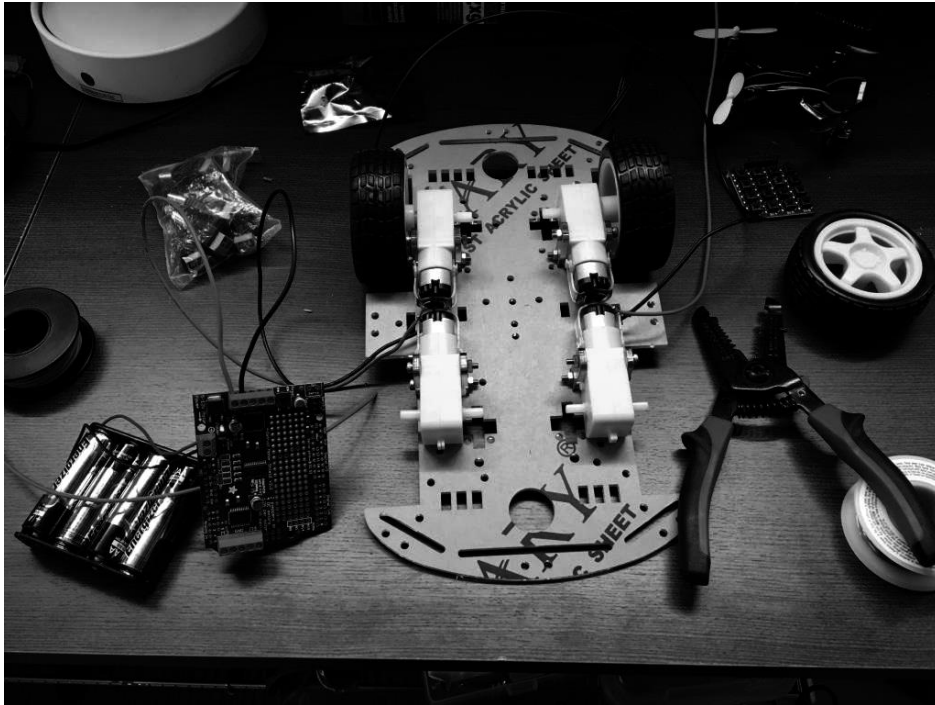
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

### **Imagen escena**

En este experimento se emplean tres imágenes escena cuya intensidad luminosa ha sido modificada de acuerdo a un valor  $\gamma$

#### **Escena ( $\gamma = 2.0$ )**

Consiste en la escena original a color tras aplicar el cambio de intensidad luminosa con  $\gamma = 2.0$ , ver Ilustración 4.4.2-1. Al iniciar el experimento, el formato de la imagen se convierte a niveles de gris.



**Ilustración 4.4.2-1: Escena con variación de intensidad uniforme  $\gamma=2.0$**

#### **Escena ( $\gamma = 1.0$ )**

Consiste en la escena original tras aplicar el cambio de intensidad luminosa con  $\gamma = 1.0$ , como puede verse en la Ilustración 4.4.2-2.



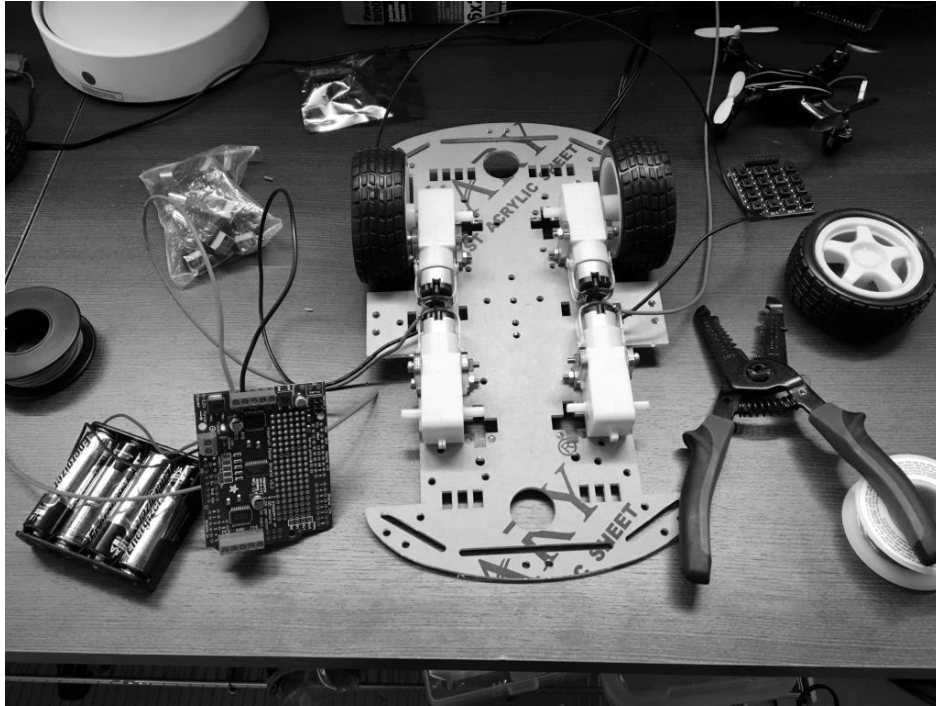


Ilustración 4.4.2-2: Escena con una variación de intensidad uniforme  $\gamma=1.0$

Aunque a simple vista pueda parecer idéntica a la imagen original, la intensidad luminosa de las mismas es distinta.

**Escena ( $\gamma = 0.5$ )**

Consiste en la escena original tras aplicar el cambio de intensidad luminosa con  $\gamma=0.5$ , atender a la Ilustración 4.4.2-3.

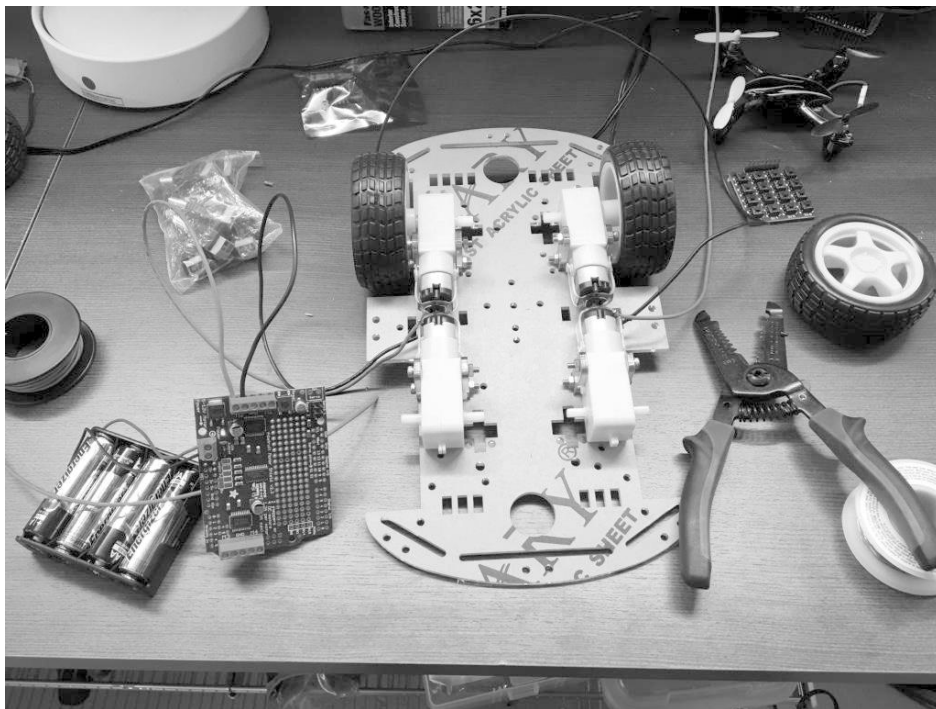


Ilustración 4.4.2-3: Escena con variación de intensidad uniforme  $\gamma=0.5$

## Plantillas

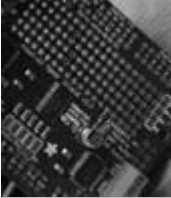


Plantilla "circuito"	Plantilla "pilas"	Plantilla "rueda"
		

Tabla 4.4.2-1: Plantillas experimento 3

En la Tabla 4.4.2-1 se pueden apreciar las plantillas del experimento. Son imágenes extraídas de la escena original, rotadas 45°, escaladas por un factor de escalado 0.7. En dicha tabla se muestran las plantillas con sus tamaños de acuerdo al escalado indicado anteriormente.

### 4.4.3. Análisis de los resultados

El análisis de los resultados se estructura de este modo: primero se narran los resultados obtenidos para cada escena y para cada plantilla en el mismo orden en que han sido presentadas escenas y plantillas. Posteriormente se contrastan los resultados generales enfatizando la capacidad de tolerancia a cambios de intensidad uniforme por parte del operador LBP y de manera secundaria se plantea qué combinaciones de parámetros DE resultan más efectivos para contribuir a dicha tolerancia a variación de la intensidad por parte del operador.

Comenzando con la escena cuyo  $\gamma=2.0$ , y la plantilla "circuito", Tabla A.3-1: Resultados influencia de la intensidad luminosa, circuito  $\gamma=2.0$ , se observa un drástico cambio de tendencia: la tasa de éxito es del 0%, en ningún caso se encuentra la solución. Las distancias intercentrales de los mejores candidatos a cada ejecución, por consiguiente superan los 50 píxeles. Cabe quizá mencionar que para la combinación  $F=0.2$  y  $CR=0.2$  dicha distancia intercentral es la menor de todas las ejecuciones situándose en 53 píxeles. Esto significa que ha quedado cerca de considerarse un "éxito".

El tiempo de ejecución por run es de aproximadamente 50 segundos; concuerda con los experimentos básicos para esta plantilla, ya que la configuración en esencia es la misma, cambiando únicamente la escena proporcionada. Esta anotación aplica del mismo modo a la revisión de los tiempos de ejecución de los demás casos escena-plantilla de este experimento por lo que no será preciso indicar nuevamente este hecho.

El MBF se sitúa en  $2 \cdot 10^{-2}$ , en cambio la desviación estándar salvo los dos primeros casos ( $F=0.2$   $CR=0.2$  y  $F=0.2$   $CR=0.5$ ) queda en un orden de magnitud de  $10^{-4}$ , lo cual parece indicar la presencia de uno o varios óptimos locales que han podido surgir precisamente debido a la aplicación de este cambio concreto con  $\gamma=2.0$  a la escena.

Los mejores genotipos de estas ejecuciones tienden a aproximarse al valor real de la dimensión del cambio de escala (1.4), las dimensiones de las coordenadas de la esquina superior izquierda de las subimágenes tienen por primera vez para este template una variabilidad no apreciada en otros experimentos, dándose los intervalos [256, 270] y [410,428] para x e y respectivamente. El ángulo de rotación nunca coincide con los 45°.

Los mejores fitness hallados, no quedan lejos de los valores fitness encontrados en otros experimentos para candidatos considerados soluciones, la mayor parte de los valores de coste son cercanos a  $2 \cdot 10^{-2}$

En el caso del template "pilas" para la escena  $\gamma=2.0$ , Tabla A.3-2: Resultados influencia de la intensidad luminosa, pilas  $\gamma=2.0$ , el valor de SR mejora levemente, ahora solo hay dos ejecuciones donde dicho SR es del 0%, en uno de ellos la distancia intercentral es completamente desproporcionada, ascendiendo a 812 píxeles. El MBF pese a lo que podría sospecharse, aumenta respecto al template anterior: en esta ocasión ronda valores cercanos a  $3 \cdot 10^{-3}$  en la mayor parte de las ejecuciones con algo más de desviación estándar, cercana ésta a  $3 \cdot 10^{-3}$ .

Dado que hay casos donde los mejores candidatos están totalmente alejados de la solución, los mejores genotipos muestran en dichos casos valores completamente aleatorios. Los mejores fitness en estos casos aunque siendo más elevados  $3 \cdot 10^{-2}$  aproximadamente, no se diferencian mucho del resto de valores fitness de esta combinación escena-template, situándose estos en torno a  $2 \cdot 10^{-2}$ .

Si atendemos a la tasa de éxito en relación a cuántas ejecuciones no encuentran ninguna solución, Tabla A.3-3: Resultados influencia de la intensidad luminosa, rueda  $\gamma=2.0$ , el caso del template "rueda" para la escena  $\gamma=2.0$ , es el template que mejores resultados ofrece para esta variación uniforme de la intensidad luminosa: solo hay una ejecución donde el SR es 0.0; aquella cuya combinación de parámetros F y CR es 0.2 en ambos casos. Incluso existe una ejecución donde el SR supera el 50%, donde la combinación de parámetros es 0.8 y 0.5 para F y CR.

Además, si atendemos las distancias intercentrales, observamos que hay tres ejecuciones donde dicha distancia es inferior a 3 píxeles. El MBF asciende, siendo en la mayor parte de estas 9 ejecuciones un valor cercano a  $10 \cdot 10^{-4}$ . La desviación estándar del MBF se sitúa en el orden de magnitud de las milésimas, pero con valores elevados. En el caso donde la distancia intercentral es 1.37 píxeles, la desviación estándar es algo mayor: 0.012. De este acontecimiento puede suponerse que el hallazgo de un candidato cercano a la solución ha sido tal vez en parte fortuito. Los mejores genotipos, en el grueso de las ejecuciones presentan valores cercanos a la solución, salvo el caso F=0.2 CR=0.2 y F=0.8 CR=0.8. Además, en estas dos ejecuciones se observa para la primera de ellas un valor del ángulo

de rotación muy desviado,  $267^\circ$  y para la segunda de ellas un valor del factor de escalado también muy desviado 0.7.

Pasando a la siguiente escena, cuyo  $\gamma=1.0$  el panorama de los resultados cambia mejorando globalmente en todos los templates.

Para el template "circuito" en Tabla A.3-4: Resultados influencia de la intensidad luminosa, circuito  $\gamma=1.0$ , la tasa de éxito es del 100% en todas las ejecuciones salvo en una donde es del 90%; precisamente en esa ejecución se halla la mejor distancia intercentral encontrada situada a 1.1 píxel del centro de la solución. Las distancias intercentrales de las demás ejecuciones están por debajo de los 7 píxeles. El MBF es algo más variable entre las ejecuciones, encontrando estos valores entre los órdenes de magnitud  $10^{-2}$  y  $10^{-3}$ , así como las desviaciones estándar de los MBF que se sitúan en los órdenes de magnitud  $10^{-2}$  y  $10^{-4}$ . Los mejores genotipos de estas ejecuciones presentan poca varianza, y tienen una precisión bastante acertada en todas las dimensiones, en especial para factor de escalado y ángulo de rotación. Se aprecia también una mejora en los costes de los mejores individuos, situándose en el intervalo [0.005, 0.007].

La búsqueda del template "pilas" en Tabla A.3-5: Resultados influencia de la intensidad luminosa, pilas  $\gamma=1.0$ , en la escena  $\gamma=1.0$ , presenta un caso con una tasa de éxito baja, 0.2, donde la combinación F CR es 0.2 para ambos casos. Los demás SR superan el 70%, uno concretamente alcanza el 100%, para F=0.8 y CR=0.2. Todas las distancias intercentrales son inferiores a 8 píxeles. El MBF tiene un orden de magnitud de centésimas en todas las ejecuciones y la desviación estándar del MBF se sitúa en el orden de magnitud  $10^{-3}$ , siendo en el caso donde el SR es 100 de un orden de magnitud menos. Las dimensiones de los mejores genotipos encontrados, mantienen la precisión para el factor de escalado y el ángulo de rotación, pero presentan una menos homogeneidad en las coordenadas x e y. Los fitness de estos genotipos oscilan en el intervalo [0.01, 0.02].

El template "rueda" presenta mejores resultados que el template "pilas" en Tabla A.3-6: Resultados influencia de la intensidad luminosa, rueda  $\gamma=1.0$ , así como mejores resultados respecto al mismo template en la escena  $\gamma=2.0$ . El SR es mínimo del 50% en todas las ejecuciones, se alcanza el 0.9 en la combinación F=0.8 CR=0.5. Las distancias intercentrales son inferiores a 6 píxeles en todas las ejecuciones salvo una. El MBF sigue estable en el orden de las centésimas, la desviación estándar del mismo se sitúa en los órdenes de magnitud  $10^{-2}$  y  $10^{-3}$ . Se observa que aumenta la amplitud de los intervalos en los que se presentan las dimensiones de los mejores genotipos encontrados, cuyos fitness pertenecen también como en el caso de las "pilas" para la esta escena  $\gamma=1.0$  al intervalo [0.01, 0.02].

La última de las escenas, donde  $\gamma=0.5$  arroja resultados diferentes para cada template.

El circuito en Tabla A.3-7: Resultados influencia de la intensidad luminosa, circuito  $\gamma=0.5$ , la tasa de éxito es nuevamente del 100% y las distancias intercentrales inferiores a 9 píxeles, incluyendo un caso donde dicha distancia es 0.9 píxeles, donde  $F=0.2$  y  $CR=0.5$ . El MBF es cercano a 0.1 en todas las ejecuciones y las desviaciones estándar de los mismos están entre  $10^{-3}$  y  $10^{-4}$  en orden de magnitud. Las dimensiones de los genotipos están próximas a la solución en todos los casos los intervalos en los que se presentan los valores de las dimensiones son estrechos. Los fitness que presentan tales genotipos se enmarcan en el intervalo [0.12, 0.14].

El template "pilas" en Tabla A.3-8: Resultados influencia de la intensidad luminosa, pilas  $\gamma=0.5$  presenta SR entre 0.4 y 0.8 para las ejecuciones de diferentes combinaciones de parámetros F y CR. Las distancias intercentrales están por debajo de los 15 píxeles. El MBF es cercano a 0.2 en la mayor parte de esta serie de ejecuciones, con desviaciones estándar en el orden de las milésimas en todos los casos. Los mejores genotipos encontrados presentan más variabilidad en los valores de sus dimensiones que el caso del template "circuito" para esta escena, quedando sus fitness dentro del intervalo [0.14, 0.23], teniendo pues más longitud que el intervalo donde se ubicaban los fitness del template "circuito".

El último caso a analizar de este experimento corresponde al template "rueda", Tabla A.3-9: Resultados influencia de la intensidad luminosa, rueda  $\gamma=0.5$ . Los resultados de SR varían desde 0.2 hasta 0.8, encontrado distancias intercentrales salvo en un caso inferiores a 12 píxeles. En dicho caso particular, la distancia es de 49 píxeles y corresponde a la combinación  $F=0.5$  y  $CR=0.8$ . De nuevo los genotipos amplían los intervalos donde se aparecen los valores de sus dimensiones, así como los fitness de los mismos que se dispersan por el intervalo [0.18, 0.28].

En general, la tolerancia a cambios de intensidad luminosa, tiene una dependencia con la clase de template que esté siendo buscado en la escena y el tipo de escena. Un template plano con intensidad elevada puede ser más fácilmente encontrado en una escena donde la intensidad tiende a ser baja. Este planteamiento tiene su origen en el análisis de los resultados obtenidos en la escena  $\gamma=2.0$ .

Otra idea que puede alcanzarse al reflexionar sobre los resultados obtenidos para la escena  $\gamma=1.0$ , es que una mejora del contraste de la escena puede facilitar la tarea de búsqueda para objetos con texturas planas.

Añadiendo en consideración los resultados obtenidos para la escena  $\gamma=0.5$ , se puede argumentar que efectivamente el operador LBP incluye como característica en su

naturaleza la tolerancia a cambios de intensidad uniforme, siempre que estos no sean excesivos.

Además, por lo observado en este experimento, parece que el rendimiento del operador sufre más cuando ese cambio de intensidad está destinado a perder intensidad. Prestando atención al template del circuito, cuyo vector de características LBP posee una cantidad relevante de ocurrencias para cada patrón LBP, se ve especialmente afectado ya que el oscurecimiento de la imagen contribuye a la pérdida de varios de esos patrones.

No obstante, tal y como se sostenía al comienzo de esta valoración global del experimento, se aprecia una cierta correlación entre el tipo de textura del template, el tipo de textura general de la escena y el cambio de intensidad uniforme aplicado.

## **4.5. Experimento 4: Influencia del ruido gaussiano**

La aplicación del ruido de tipo gaussiano, consiste en probabilísticamente variar píxeles que componen la imagen para cambiar su intensidad, de acuerdo con una distribución normal o gaussiana. La curtosis de la distribución se determina especificando una media y un valor de desviación,  $\gamma$ . En este experimento se emplea media 0 y diferentes valores  $\gamma$ .

### **4.5.1. Objetivo**

1. Comprobar la tolerancia del operador LBP al ruido gaussiano
2. De forma secundaria, se observa el comportamiento de las combinaciones de parámetros DE, F y CR en relación a la aplicación de ruido gaussiano.

### **4.5.2. Configuración del Experimento**

Aplicación de una variación de ruido gaussiano,  $\gamma = 10^{-3}$  a una escena original

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

Aplicación de una variación de ruido gaussiano,  $\gamma = 10^{-2}$  a una escena original

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

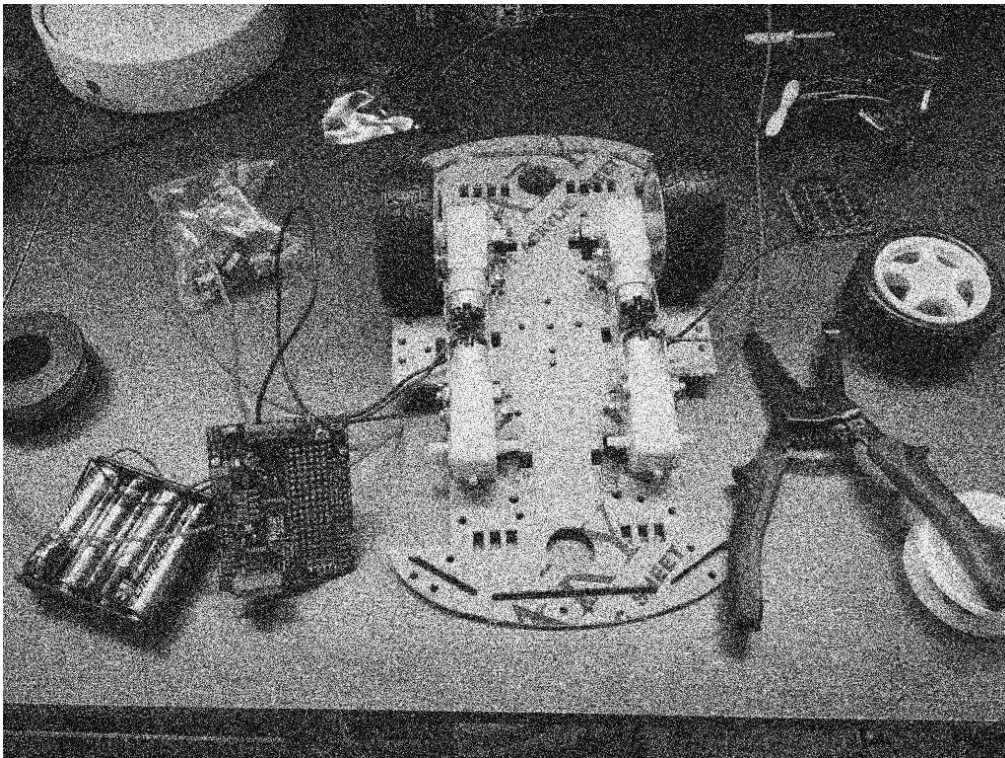
Aplicación de una variación de ruido gaussiano con  $\gamma = 10^{-1}$  a una escena original

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

### Imagen escena

En este experimento se emplean tres imágenes escena que proceden de una imagen "origen" cuya diferencia reside en valor  $\gamma$  de un filtro gaussiano aplicado a la imagen origen.

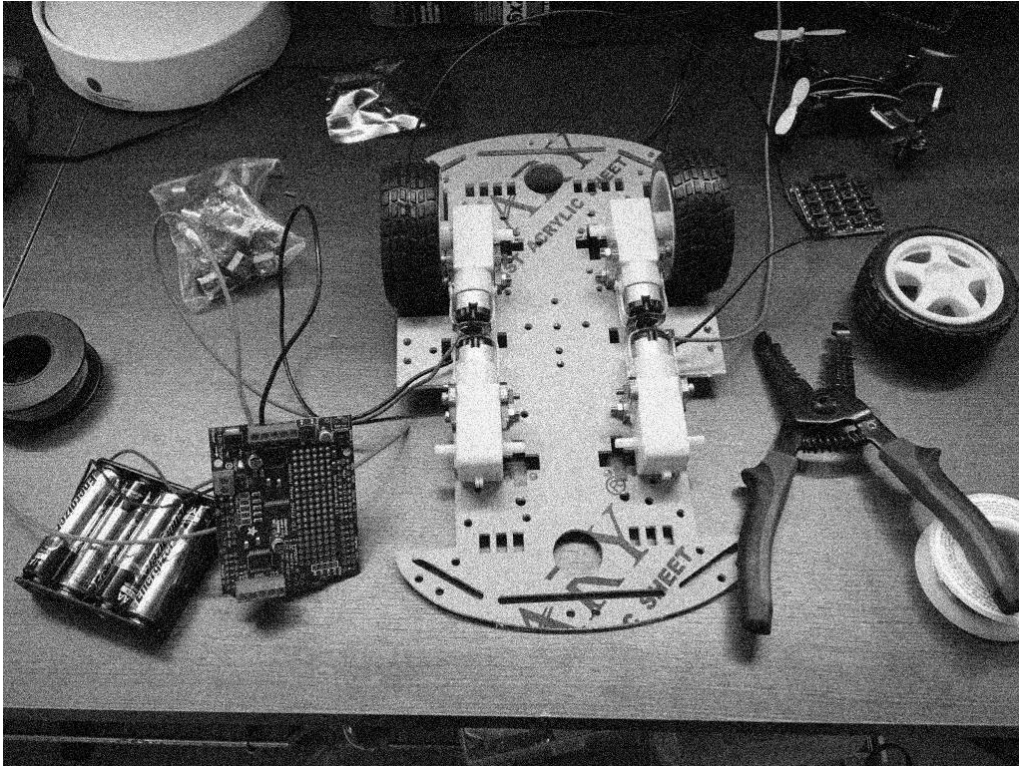
#### Escena gaussiana $\gamma = 10^{-1}$



**Ilustración 4.5.2-1: Escena experimento 4,  $\gamma=0.1$**

Puede apreciarse la gran cantidad de contaminación introducida por el ruido gaussiano en la Ilustración 4.5.2-1 para un valor cercano a 1 de  $\gamma$ .

Escena gaussiana  $\gamma = 10^{-2}$



**Ilustración 4.5.2-2: Escena experimento 4,  $\gamma=0.01$**

En este caso, el valor  $\gamma$  introduce un ruido más modesto, como puede apreciarse en la Ilustración 4.5.2-2.



## Escena gaussiana $\gamma = 10^{-3}$

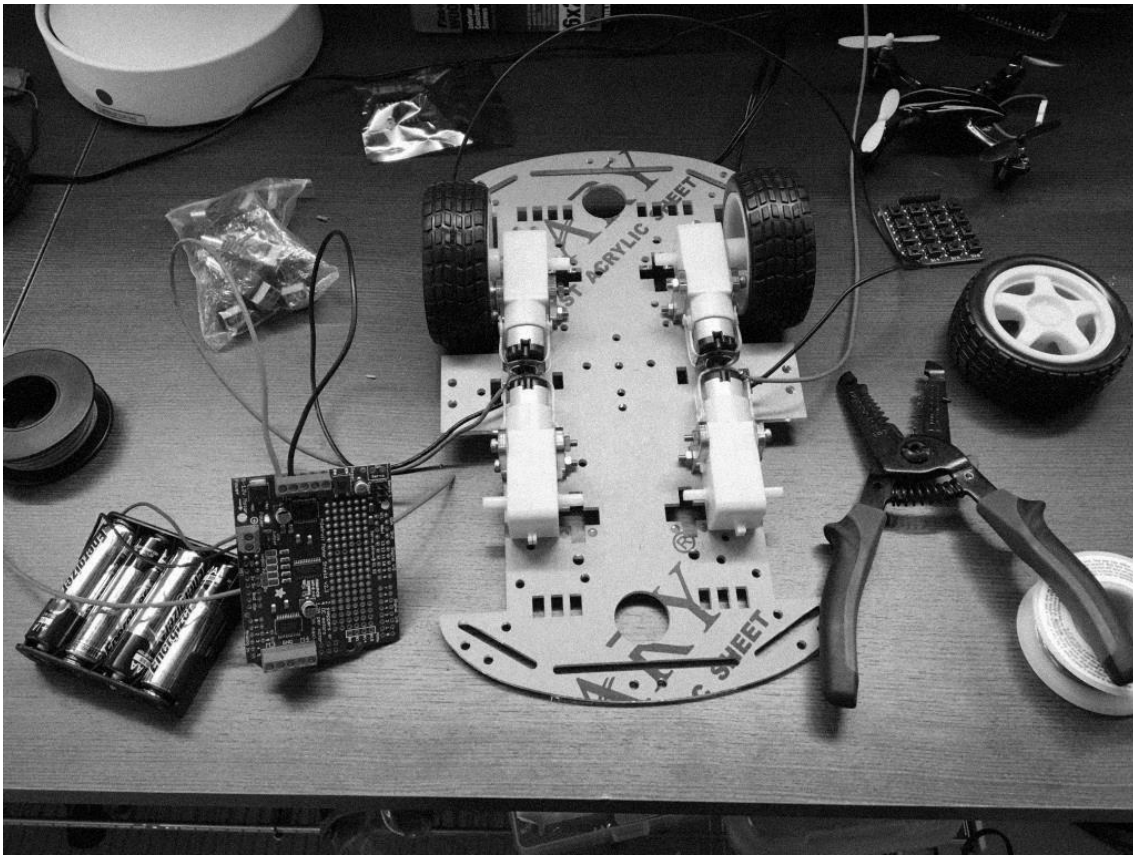


Ilustración 4.5.2-3: Escena experimento 4,  $\gamma=0.001$

En la Ilustración 4.5.2-3, el ruido ya es casi imperceptible para  $\gamma$  muy pequeño.

### Templates

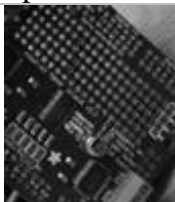


Template "circuito"	Template "pilas"	Template "rueda"
		

Tabla 4.5.2-1: Templates experimento 4

En la Tabla 4.5.2-1, se pueden apreciar los templates del experimento. Son imágenes extraídas de la escena original, rotadas  $45^\circ$ , escaladas por un factor de escalado 0.7 y convertidas a niveles de gris. En dicha tabla se muestran los templates con sus tamaños de acuerdo al escalado indicado anteriormente.

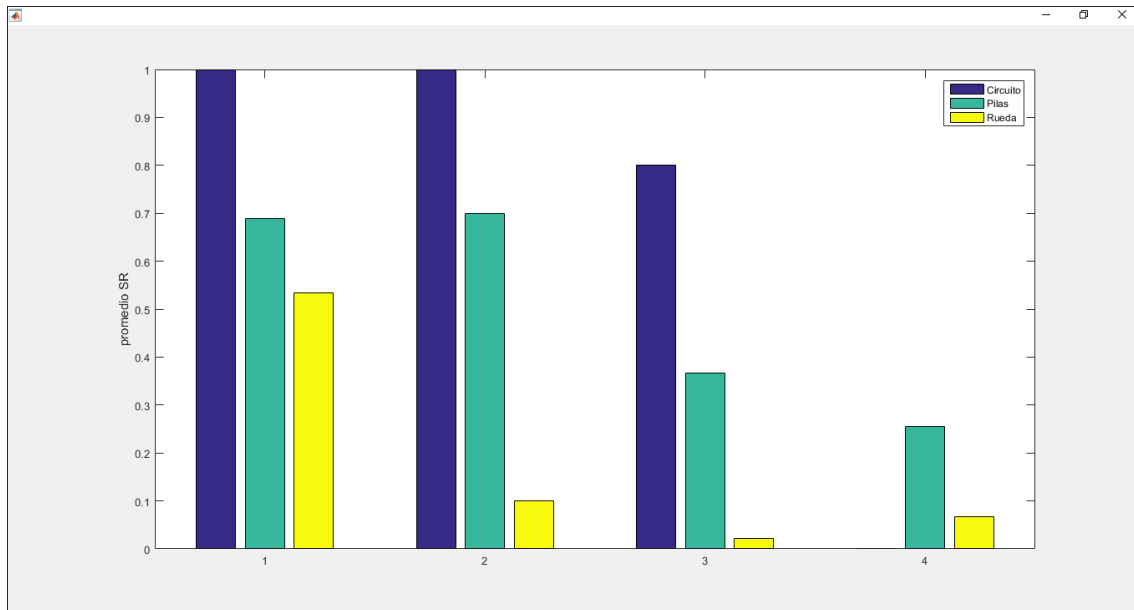
### 4.5.3. Análisis de los resultados

A continuación se reflexiona sobre los resultados presentándose dichas reflexiones en este orden. Primero se analizará cada par escena-template por separado, añadiendo comparaciones relevantes entre las mismas. A continuación se expondrán los resultados generales para cada tipo de escena y por último se valora la capacidad de tolerancia de este tipo de ruido por parte del operador LBP.

La primera escena donde  $\gamma=10^{-3}$ , Tabla A.4-1, Tabla A.4-2, Tabla A.4-3; el circuito se parece no ser afectado por el ruido ya que alcanza el éxito total en todas las ejecuciones. El template rueda alcanza un SR superior al 60% en casi todas las ejecuciones, salvo para la combinación F y CR de 0.8 ambos. La rueda alcanza un SR de 40% como máximo.

En la segunda escena, consultar resultados en Tabla A.4-7, Tabla A.4-6 y Tabla A.4-5, con 0.01 para el valor  $\gamma$ , el template circuito alcanza el SR 100% en dos tres ejecuciones. Puede observarse que todas las ejecuciones tienen para la mejor solución encontrada distancias intercentrales menores a 33 píxeles. El MBF se sitúa en torno al 0.25, en el caso del template pilas, el SR llega al 80% en una ejecución. El template rueda pierde rendimiento, llega solo al 10% de SR en dos ejecuciones.

Finalmente la escena cuyo  $\gamma=10^{-1}$ , ver resultados en Tabla A.4-7, Tabla A.4-8, y Tabla A.4-9: contemplamos que el caso del template "circuito" no alcanza en ningún caso una ejecución exitosa. Las distancias intercentrales rondan los 800 píxeles. El MBF se sitúa entre los valores 0.2 y 0.3 y los mejores fitness se sitúan en torno al valor 0.03. En el caso del template pilas, si se encuentran soluciones en varias ejecuciones, pero el SR nunca supera el 50%. Para el template rueda hay tres ejecuciones donde el SR llega al 20%, en las demás es nulo.



**Figura 4.5.3-1: Pérdida de rendimiento Ruido Gaussiano**

En la Figura 4.5.3-1 podemos observar 4 grupos de valores promediados de SR. El primer grupo corresponde a los resultados promediados de SR obtenidos de: Tabla A.1-4: Resultados decodificación básica TLFSR, Tabla A.2-1: Resultados exploración templates homogéneos, pilas GR=250, y Tabla A.2-2: Resultados exploración templates homogéneos, rueda GR=250. Los siguientes grupos, 2, 3 y 4 corresponden a los valores promediados de cada template obtenidos para este experimento. El segundo grupo corresponde a  $\gamma=0.001$ , el tercero a  $\gamma=0.2$  y el cuarto a  $\gamma=0.1$ .

Se hace tangente la pérdida de rendimiento según el valor  $\gamma$  se aproxima a 1.

En términos generales, es especialmente llamativa pérdida total de la capacidad de encontrar el template circuito en una escena con un ruido gaussiano alto, ya que este template solía ser encontrado en las demás circunstancias.

Por lo demás, como cabía esperar se aprecia una pérdida de rendimiento según se incrementa el nivel de ruido. De nuevo depende en gran medida del tipo de template buscado en la escena.

## 4.6. Experimento 5: Influencia del ruido sal y pimienta

La aplicación de este tipo de ruido a una imagen digital, consiste en variar la intensidad de un porcentaje de píxeles de la escena de acuerdo a un parámetro  $\gamma$ . Es decir,  $\gamma$  se emplea

para calcular el número de píxeles cuya intensidad se modificará para ser máxima o mínima. Los píxeles modificados seguirán una distribución normal.

#### **4.6.1. Objetivo**

1. Comprobar la tolerancia del operador LBP al ruido sal y pimienta "*Salt&Pepper*"
2. De forma secundaria, se observa el comportamiento de las combinaciones de parámetros DE, F y CR en relación a la aplicación de ruido "*Salt&Pepper*".

#### **4.6.2. Configuración del Experimento**

Aplicación de una variación de ruido "*Salt&Pepper*" con  $\gamma = 0.01$  a una escena origen

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

Aplicación de una variación de ruido "*Salt&Pepper*" con  $\gamma = 0.05$  a una escena origen

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

Aplicación de una variación de ruido "*Salt&Pepper*",  $\gamma = 0.1$  a una escena origen

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

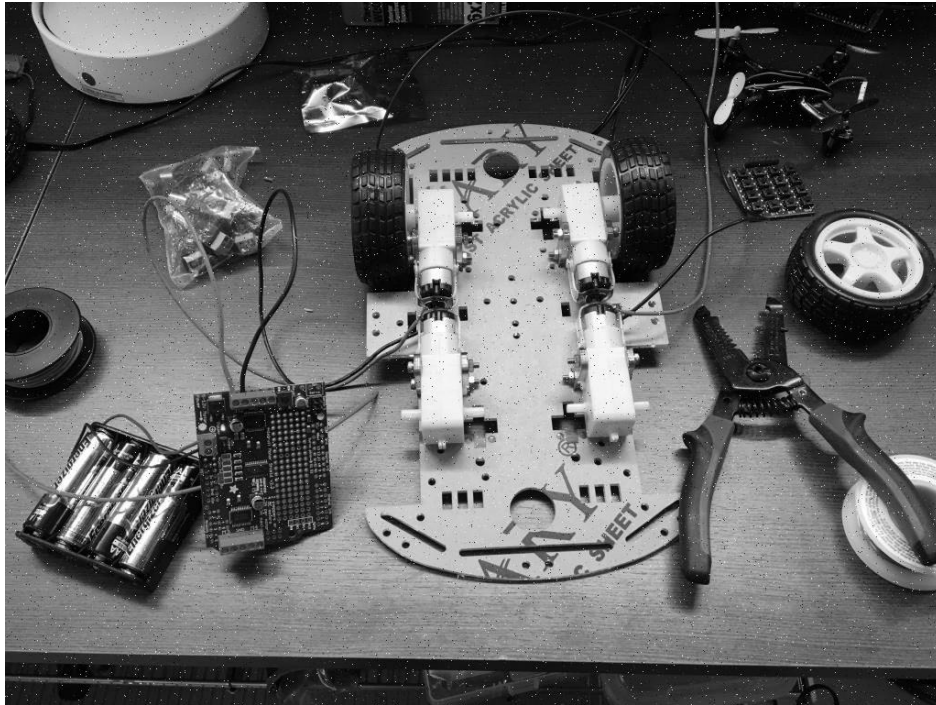
#### **Imagen escena**

En este experimento se emplean tres imágenes escena que proceden de una imagen "origen". La diferencia de las imágenes derivadas reside en el valor del parámetro de entrada  $\gamma$  asignado al filtro sal y pimienta aplicado para obtener dichas escenas derivadas.

La cantidad artefactos que genera sobre una imagen la aplicación de este filtro, aumenta cuanto mayor sea el parámetro  $\gamma$ , sustituyendo un porcentaje de píxeles de la escena por valores de intensidad máximos y mínimos, distribuyéndolos por la imagen siguiendo una distribución normal.

### **Salt&Pepper ( $\gamma = 0.01$ )**

En la Ilustración 4.6.2-1 pueden apreciarse sobre todo píxeles con intensidades máximas distribuidos por la escena.



**Ilustración 4.6.2-1: Escena con ruido sal y pimienta  $\gamma=0.01$**

### **Salt&Pepper ( $\gamma = 0.05$ )**

En la Ilustración 4.6.2-2, el ruido es perfectamente visible en toda la imagen.

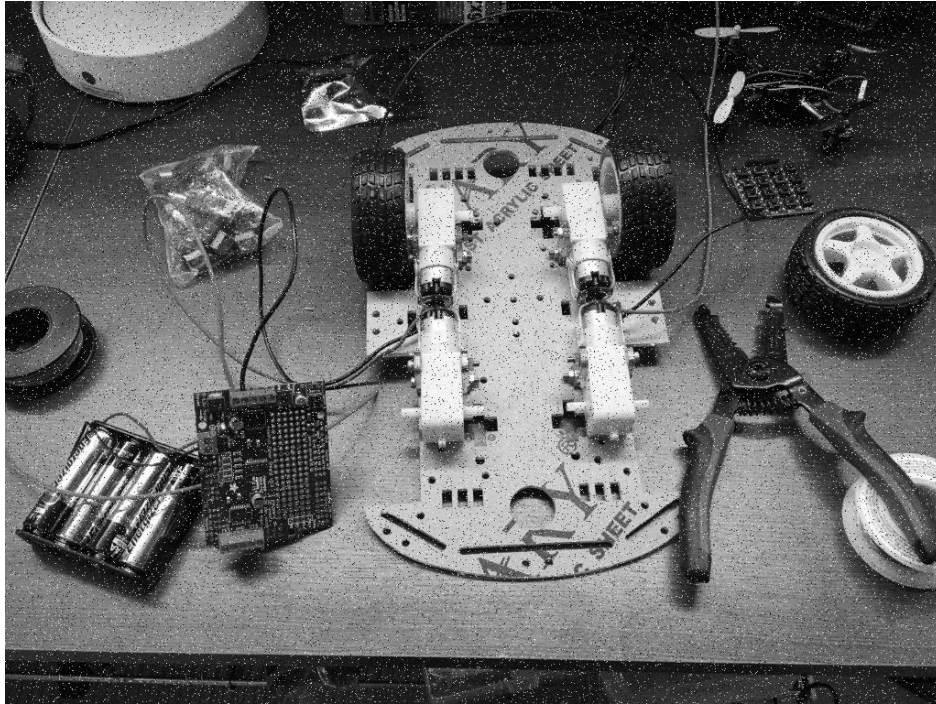


Ilustración 4.6.2-2: Escena con ruido sal y pimienta  $\gamma=0.05$

**Salt&Pepper ( $\gamma = 0.1$ )**

En la Ilustración 4.6.2-3, el ruido aplicado es tal que incluso el ojo humano comienza a tener dificultades para discernir ciertas regiones.

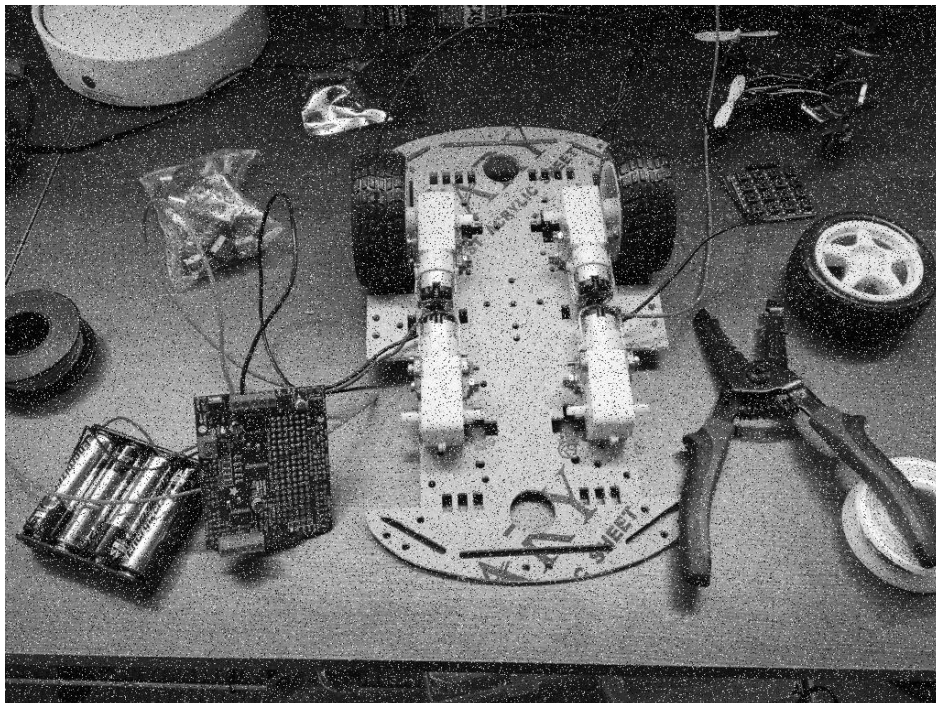


Ilustración 4.6.2-3: Escena con ruido sal y pimienta  $\gamma=0.1$

### 4.6.3. Análisis de los resultados

El experimento se analiza de la siguiente forma. Primero de manera aislada, se hace una revisión de los resultados de cada pareja escena y template. Posteriormente se analiza el impacto en el rendimiento del operador LBP que supone la aplicación de distintos niveles de ruido Sal y Pimienta y por último se contrasta el impacto del filtro en relación a los distintos templates.

En la primera tanda de resultados, Tabla A.5-1: Resultados influencia ruido sal y pimienta, circuito  $\gamma=0.01$ , el SR no desciende del 90% y las distancias intercentrales no lo hacen de 12 píxeles. El MBF medio de las distintas ejecuciones varía entre 0.008 y 0.012, manteniendo desviaciones estándar de entre  $10^{-2}$  y  $10^{-4}$  órdenes de magnitud. La duración media del run es similar al caso de exploración paramétrica para este template ya que la decodificación y asignación de coste coinciden. Lo mismo aplica para el resto de casos del experimento, por lo que no se insistirá en observar la duración de los runs. Los mejores genotipos presentan valores en sus dimensiones bastante cercanos en general, sin anomalías. Sus *fitness* pertenecen al intervalo [0.0075, 0.0098].

Continuando con la escena *Salt&Pepper* ( $\gamma = 0.01$ ) y observando los resultados para el template "pilas" en Tabla A.5-2: Resultados influencia ruido sal y pimienta, pilas  $\gamma=0.01$  el SR desciende, siendo siempre superior al 40% y no alcanzando el éxito total en ninguna ejecución. Las distancias intercentrales varían desde los 4 píxeles hasta los 22. Los valores de MBF se localizan en el intervalo [0.018, 0.042] con desviaciones de magnitudes de milésimas o centésimas. La variabilidad de los genotipos se acentúa, encontrando un caso que presenta una coordenada x muy alejada del resto de los valores hallados para la misma dimensión en los demás genotipos. Los fitness se sitúan entre los valores 0.010 y 0.023.

El caso del template "rueda" Tabla A.5-6: Resultados influencia ruido sal y pimienta, rueda  $\gamma=0.05$ , presenta una tasa de éxito de entre 0.3 y 0.7. Las distancias intercentrales son algo más dispersas, siendo la menor de 4 píxeles y la mayor de 38. El MBF de cada ejecución pertenece al intervalo [0.029, 0.042] todas las desviaciones estándar tienen un orden de magnitud de  $10^{-3}$ . En este caso, los genotipos son bastante similares en las coordenadas x e y, pero el factor de escalado queda algo distante del Ground Truth, siendo el mejor caso de 1.35 respecto al 1.44 del Ground Truth. El ángulo de rotación si es  $45^\circ$  para uno de los genotipos, sin embargo también hay bastante varianza en esta dimensión. El mejor coste tiene valor 0.017 y el menos óptimo 0.031.

Observando la escena *Salt&Pepper* ( $\gamma = 0.5$ ) y el template "circuito", Tabla A.5-4: Resultados influencia ruido sal y pimienta, circuito  $\gamma=0.05$ , se aprecia que el SR desciende en una ejecución al 80%, y las distancias intercentrales aumentan quedando entre 17 y 27 píxeles. Los MFB de estas ejecuciones, están comprendidas en el intervalo [0.018, 0.021] con desviaciones estándar nuevamente de centésimas o milésimas. Los genotipos parecen

perder precisión sobre todo en la dimensión del cambio de escala, las coordenadas x e y son muy similares y el ángulo de rotación no se aleja de los 45°. Los costes se sitúan en el intervalo [0.015, 0.020].

El template "pilas" junto a la escena *Salt&Pepper* ( $\gamma = 0.5$ ), ver Tabla A.5-5: Resultados influencia ruido sal y pimienta, pilas  $\gamma=0.05$ , sufre una pérdida de eficacia cuando es analizada la tasa de éxito: es 0 en numerosas ejecuciones y no supera el 40% en ningún caso. Las distancias intercentrales rondan los 55 píxeles en todas las ejecuciones. Los MBF se localizan en el intervalo [0.042, 0.056] con un orden de magnitud en sus desviaciones estándar de  $10^{-3}$ . En los genotipos se parecía una cierta homogeneidad con el inconveniente de que los valores de la dimensión del cambio de escala son justo la inversa del Ground Truth: 0.7. Los fitness oscilan entre los valores 0.031 y 0.043.

Para el caso de la rueda, Tabla A.5-6: Resultados influencia ruido sal y pimienta, rueda  $\gamma=0.05$ , y continuando con la escena *Salt&Pepper* ( $\gamma = 0.5$ ) la pérdida de eficacia del operador es todavía más acusada que para el template anterior, teniendo éxitos solo en tres ejecuciones con una tasa máxima del 20%. Las distancias entre los centros del hipotético mejor individuo y la solución llegan casi hasta los 500 píxeles en una ejecución. El MBF se sitúa entre los valores 0.042 y 0.050 manteniéndose las desviaciones estándar de los mismos en orden de magnitud  $10^{-3}$ . En los genotipos se detecta que ni la dimensión del factor de escalado ni la que representa el ángulo de rotación se aproximan en ningún caso a los valores de la solución real. Este último hecho junto hay que añadirle una muy notable varianza (de cientos de píxeles) entre las coordenadas x e y. Los costes, se sitúan en el intervalo [0.038, 0.046].

En la escena *Salt&Pepper* ( $\gamma = 0.1$ ) junto al template "circuito", el SR ya no alcanza el 100% en ninguna ejecución: Tabla A.5-7: Resultados influencia ruido sal y pimienta, circuito  $\gamma=0.1$  la tasa de éxito se mantiene entre 0.5 y 0.8. Las distancias intercentrales son inferiores a 49 píxeles, llegando en una ocasión hasta los 736. Los MBF por su parte quedan comprendidos en el intervalo [0.023, 0.027]. Las desviaciones típicas de los MBF presentan órdenes de magnitud de  $10^{-3}$  y  $10^{-4}$ . Los genotipos exhiben una dispersión elevada muy poco habitual para este template, en todas las dimensiones. Los costes se encuentran en el intervalo [0.022, 0.026].

El template de las pilas en compañía de la escena *Salt&Pepper* ( $\gamma = 0.1$ ), Tabla A.5-8: Resultados influencia ruido sal y pimienta, pilas  $\gamma=0.1$ , alcanza como máximo una tasa de éxito de 0.3, siendo 0 en casi la mitad de las ejecuciones. Las distancias de los centros halladas distan de unos 50 o 60 píxeles. El MBF ocupa el intervalo [0.059, 0.053] para estas ejecuciones, con desviaciones estándar de  $10^{-2}$  y  $10^{-3}$  en orden de magnitud. Los genotipos son visiblemente más homogéneos que en el anterior template para esta escena; la dimensión del ángulo de rotación es cercana a los 45° en casi todos los candidatos pero



la dimensión del factor de escalado de nuevo es inversa al valor del Ground Truth. Los fitness se localizan en el intervalo [0.051, 0.065].

El template "rueda" en esta escena, Tabla A.5-8: Resultados influencia ruido sal y pimienta, pilas  $\gamma=0.1$ , presenta una tasa de éxito similar a la del template de las pilas, con una máxima de SR de 30% y cuatro ejecuciones con SR de 0%. Atendiendo a las distancias intercentrales encontramos que abundan distancias que superan los 500 píxeles, por el contrario la mejor distancia es de 29 píxeles. Los MBF están comprendidos en el intervalo [0.050, 0.058], con desviaciones estándar todas en el orden de magnitud de las milésimas. Contemplando los genotipos encontramos que la varianza es elevada para las coordenadas x e y, el factor de escalado de nuevo se aleja considerablemente del Ground Truth y además los ángulos de rotación solo en dos casos se aproximan a  $45^\circ$ . Los costes aparecen comprendidos en el intervalo [0.045, 0.045].

A nivel global, es posible argumentar que la tolerancia a ruido de tipo sal y pimienta por parte del operador LBP, es reducida. Se ha observado una correlación entre el deterioro del rendimiento del operador y el aumento del parámetro  $\gamma$  en líneas generales independientemente del template empleado, aunque dicha pérdida de rendimiento ha sido más acentuada para el template de la rueda.

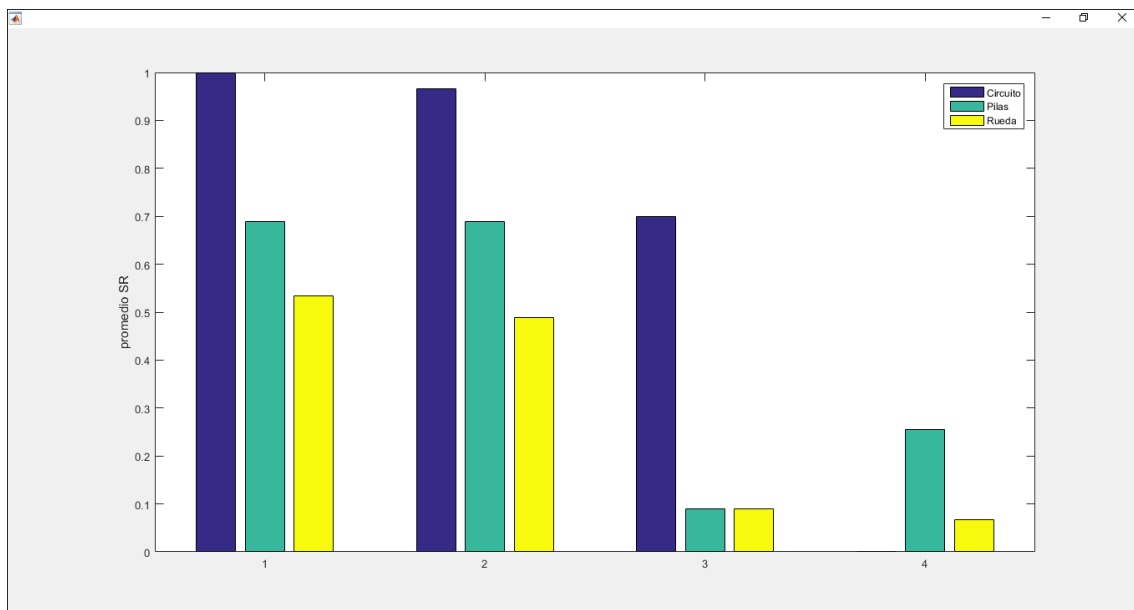


Figura 4.6.3-1: Pérdida de rendimiento sal y pimienta

En la Figura 4.6.3-1 se observan cuatro bloques de valores promediados del SR para cada uno de los templates. El primero corresponde a los valores obtenidos en ausencia de ruido. El segundo bloque corresponde a los SR promediados cuando el parámetro  $\gamma=0.01$ , el tercero cuando  $\gamma=0.1$  y el último bloque cuando  $\gamma=0.5$ .

No solo la propia naturaleza interna del template "circuito", en el sentido de composición de características LBP, sino que teniendo en cuenta la escena en la que se ubica este template, han contribuido a que la influencia del ruido no haya sido tan crítica en este caso. Los otros dos templates, por las mismas razones han sufrido una mayor incapacidad de la metodología para encontrar soluciones.

A la revisión de los resultados, es factible suponer que dependiendo de la naturaleza del template y la escena, la aplicación de este tipo de ruido para el contexto de la metodología empleada donde se emplea el histograma del operador LBP como medida de comparación, puede contribuir a la aparición de óptimos locales.

## 4.7. Experimento 6: Influencia del color y decodificación para imágenes RGB

RGB es un modelo de color basado en la síntesis aditiva, con el que es posible representar un color mediante la mezcla por adición de los tres colores de luz primarios. Las imágenes en RGB tienen un canal de intensidades para cada color.

### 4.7.1. Objetivo

Los objetivos esenciales de este experimento son:

1. Examinar si determinar el coste del individuo en base a los histogramas LBP de cada canal RGB contribuye a la precisión de la metodología.
2. Contrastar los resultados de las distintas combinaciones de parámetros DE entre los distintos templates empleados y respecto a experimentos anteriores.

Para compatibilizar la metodología con imágenes a color, ha sido necesario introducir modificaciones en la función de evaluación, como se indica en la Ecuación 4.7.1-1:

$$f = \frac{\text{dist}(\text{hist}(\text{LBP}(t_R)) - \text{hist}(\text{LBP}(i_R))) + \text{dist}(\text{hist}(\text{LBP}(t_G)) - \text{hist}(\text{LBP}(i_G))) + \text{dist}(\text{hist}(\text{LBP}(t_B)) - \text{hist}(\text{LBP}(i_B)))}{r}$$

**Ecuación 4.7.1-1: Función Fitness para imágenes a color**

- $t_R$  es la matriz imagen (canal rojo de RGB) del template
- $i_R$  es la matriz imagen (canal rojo de RGB) del individuo
- $t_G$  es la matriz imagen (canal verde de RGB) del template
- $i_G$  es la matriz imagen (canal verde de RGB) del individuo
- $t_B$  es la matriz imagen (canal azul de RGB) del template
- $i_B$  es la matriz imagen (canal azul de RGB) del individuo

- *dist* es la distancia Euclídea, que en este caso corresponde entre el vector de características LBP del template y el vector de características LBP del individuo.
- *LBP* es el operador LBP
- *hist* referencia el vector de características resultante de la aplicación del operador LBP
- *r* es el resto, obtenido a través de la Ecuación 3.2.3-2

#### 4.7.2. Configuración del Experimento

Para este experimento se realizan 27 ejecuciones distribuidas del siguiente modo:

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

#### Imagen escena

La siguiente imagen es la escena original del experimento, en esta ocasión la imagen escena no se transforma a niveles de gris, según Ilustración 4.7.2-1.

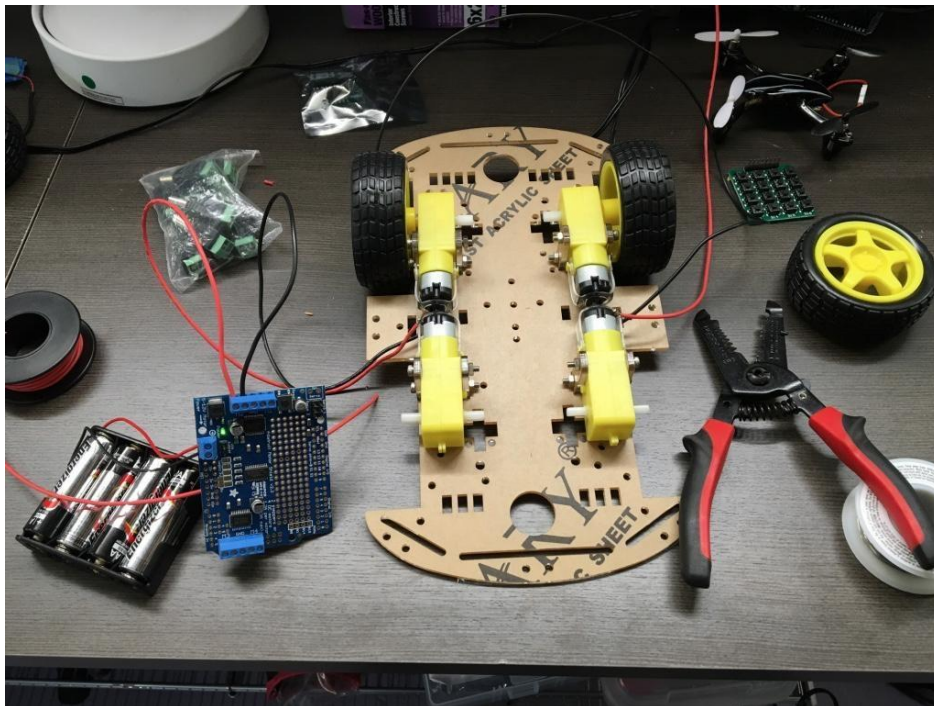


Ilustración 4.7.2-1: Escena a color

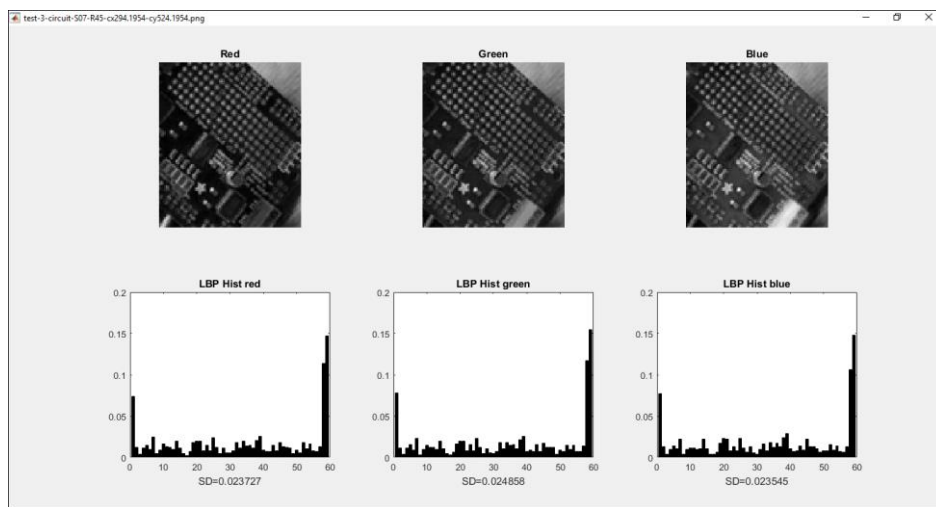
## Templates

Todos los templates de este experimento se orientan al tipo de individuo "TLFSR" a los cuales les han sido aplicadas las mismas transformaciones geométricas tanto para el cambio de escala (0.7) como para la rotación (45°), pero estos templates no son transformados a imágenes en niveles de gris.

A continuación se presentan los templates empleados junto a sus histogramas LBP normalizados de cada uno de los canales RGB para cada template.

### Template circuito

En la Ilustración 4.7.2-2 podemos apreciar las imágenes de los canales RGB de un template así como sus respectivos histogramas LBP.

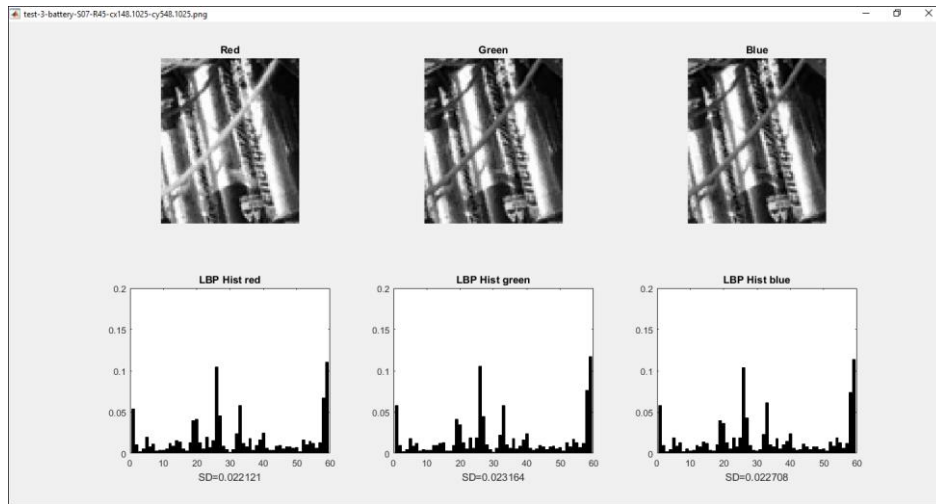


**Ilustración 4.7.2-2: Vista de los canales RGB del template circuito y el histograma LBP de cada canal**

No existen diferencias notables entre los canales RGB en cuanto a la distribución de las características LBP.

### Template pilas

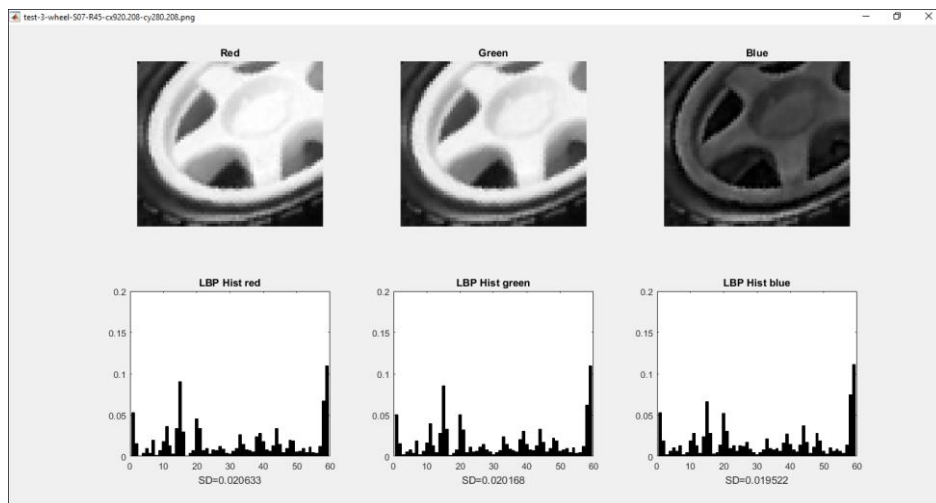
En Ilustración 4.7.2-3, se muestran las diferentes imágenes en gris para cada canal RGB del *template* pilas y sus respectivos histogramas LBP.



**Ilustración 4.7.2-3: Vista de los canales RGB del template pilas y el histograma LBP de cada canal**

El template "pilas". Cada columna en la figura anterior se corresponde a cada uno de los canales RGB (rojo, verde, azul). No se aprecian grandes diferencias entre los mismos.

### Template rueda



**Ilustración 4.7.2-4: Vista de los canales RGB del template rueda y el histograma LBP de cada canal**

En el caso del template "rueda", Ilustración 4.7.2-4, sí que se aprecian diferencias, esencialmente en las imágenes del nivel de gris de cada canal. La rueda es predominantemente amarilla y negra, y el impacto que tiene esa predominancia de una textura amarilla bastante uniforme es considerable.

Cabe destacar que pese a apreciar esas diferencias en las imágenes, la distribución de los histogramas LBP de cada canal RGB no varía tan drásticamente. Después de todo, el operador LBP está enfocado al análisis de los valores de intensidad de los vecinos de cada píxel respecto a un píxel central: aunque cambia la "intensidad" de cada color en cada canal, esta variación tiende a ser más o menos constante.

### 4.7.3. Análisis de los resultados

Los resultados de este experimento se realizan del siguiente modo. Primero se analizan los resultados de cada template por separado, a continuación se discute el comportamiento general del modelo de decodificación y coste empleado según los resultados obtenidos.

Los mejores resultados son los del template "circuito" en Tabla A.6-1: Resultados influencia del color, circuito donde la tasa de éxito (SR) es del 100% en todos los casos, con distancias intercentrales inferiores a 6 píxeles. El MBF así como su desviación estándar se mantienen en niveles del orden de magnitud de  $10^{-3}$  y  $10^{-4}$  respectivamente, sin apreciarse apenas diferencias para las distintas combinaciones de parámetros.

El tiempo de ejecución, como cabía esperar asciende considerablemente para este modelo de decodificación, alcanzando los 96 segundos por run de media y unos 960 segundos por ejecución de media, es decir algo más de 15 minutos.

Puede observarse que los mejores genotipos encontrados para cada ejecución se aproximan considerablemente a la solución, especialmente en las dimensiones que representan el factor de escalado y el ángulo de rotación, con costes (de individuo) inferiores  $10^{-2}$  en todo momento.

Observando los resultados obtenidos para el template "pilas" Tabla A.6-2: Resultados influencia del color, pilas, en cambio, se detecta una disminución drástica del SR respecto al anterior template, donde el peor resultado de SR es del 50% y encontrando un único caso donde la tasa de éxito es del 100%, que coincide con la combinación de parámetros DE F=0.8 y CR=0.2. Las distancias intercentrales, salvo en el caso de la combinación de parámetros DE F=0.8 y CR=0.2, son inferiores a 7 píxeles también, lo cual hace suponer que aunque sea más costoso encontrar la solución, se terminan encontrando individuos cercanos a ella en todas las ejecuciones.

Se puede apreciar que para este template, el MBF asciende, la mayor parte de los casos se sitúan en un orden de magnitud de  $10^{-2}$ , y las respectivas desviaciones estándar se sitúan en el orden de magnitud  $10^{-3}$ , indicando mayor variabilidad entre las soluciones encontradas y peor calidad. El tiempo de ejecución medio del run está en los 77 segundos, y el de ejecución en unos 770. Es menor que en el caso anterior básicamente porque el template también tiene un tamaño menor.

Las diferencias entre los mejores genotipos encontrados para cada ejecución, también son apreciables. La única dimensión que parece estar más cerca de la solución en términos generales es la del ángulo de rotación. No hay ningún mejor fitness inferior a  $10^{-2}$

El template "rueda", Tabla A.6-3: Resultados influencia del color, rueda; todavía arroja unos resultados menos óptimos que el anterior. Ninguna ejecución alcanza una tasa de

éxito del 100%, en el mejor de los casos el SR alcanza el valor 0.8. Las distancias intercentrales se mantienen por debajo de los siete píxeles en todo momento. De nuevo, encontrar una solución es menos probable pero siguen encontrándose soluciones buenas a nivel de ejecución. El MBF continúa aumentando citándose aproximadamente en  $3 \cdot 10^{-2}$  en términos generales para cada ejecución. La desviación estándar del MBF presenta mayor varianza entre las distintas ejecuciones de este experimento.

El tiempo de ejecución por run ronda el minuto, este template es el de menor tamaño. Los mejores genotipos correspondientes a cada ejecución son dispares. La coordenada x se sitúa en un amplio intervalo [855, 864], la coordenada y se enmarca en el intervalo [217, 228], el factor de escalado se sitúa cercano a 1.33 en la mayoría de los casos y el ángulo de rotación nunca coincide con los  $45^\circ$  de la solución.

Los mejores fitness de cada ejecución generalmente son cercanos a  $2 \cdot 10^{-2}$ , por lo tanto se puede concluir que las soluciones de este template son las menos óptimas para este experimento.

En general, se puede concluir que este tipo de decodificación no ofrece mejoras respecto a la decodificación básica. La Figura 4.7.3-1, muestra una comparativa de los valores SR obtenidos en este experimento para los templates pilas y rueda respecto a los obtenidos para los mismos templates empleando la función fitness normal en el Experimento 2.

No hay una clara mejora general el SR por considerar los canales RGB.

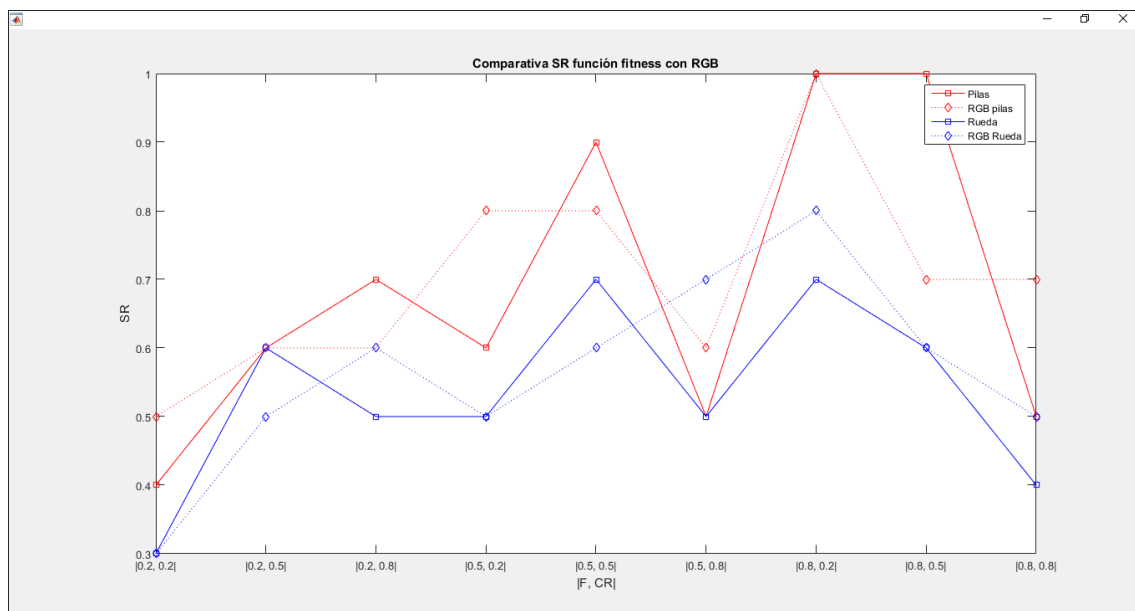


Figura 4.7.3-1: Comparación SR decodificación RGB

Aunque se siguen encontrando soluciones cercanas en cada ejecución, la probabilidad de encontrarlas decrece significativamente respecto a los experimentos realizados con la decodificación básica.

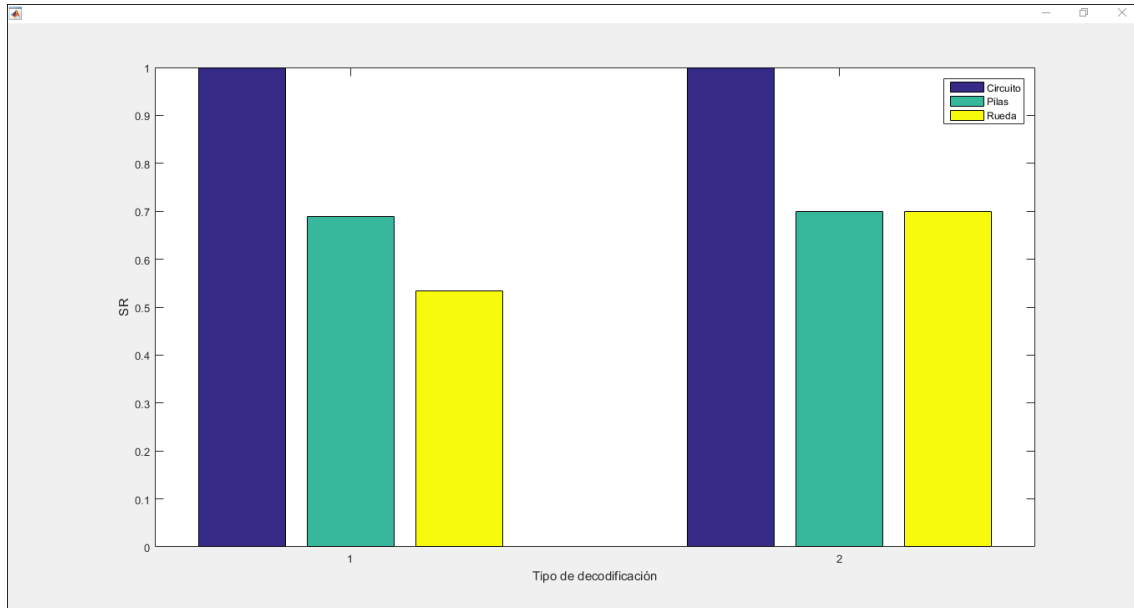


Figura 4.7.3-2: Comparación del promedio SR gris y color

En la Figura 4.7.3-2, se pueden comparar los promedios de SR para cada template en la decodificación básica de niveles de gris (grupo 1, de la izquierda) respecto al promedio de SR para imágenes a color (grupo 2, de la derecha).

El balance general conduce a concluir que esta aproximación no proporciona beneficios al método; amplía el tiempo de ejecución, no aumenta la tasa de éxito y termina con candidatos a solución de menor calidad respecto al mismo experimento con decodificación básica, basada en imágenes de niveles de intensidad en gris.

## 4.8. Experimento 7: Combinación de diferentes Operadores LBP

Para este experimento se plantea una función fitness alternativa, según Ecuación 4.7.3-1:

$$f = \frac{\text{dist}(\text{hist}(LBP_1(t)) - \text{hist}(LBP_1(i))) + \text{dist}(\text{hist}(LBP_2(t)) - \text{hist}(LBP_2(i)))}{r}$$

Ecuación 4.7.3-1: Función Fitness promedio de dos LBP

Donde:

- $t$  es la matriz imagen (escala de grises) del template
- $i$  es la matriz imagen (escala de grises) del individuo
- $\text{dist}$  es la distancia Euclídea, que en este caso corresponden entre el vector de características LBP del template y el vector de características LBP del individuo.
- $LBP_1$  es el operador LBP
- $LBP_2$  es el operador LBP



- $LBP_1$  y  $LBP_2$  utilizan diferente combinación de parámetros R y P
- R es el parámetro radio del operador LBP
- P es el parámetro que indica el tamaño del vecindario del operador LBP
- *hist* referencia el vector de características resultante de la aplicación del operador LBP
- *r* es el resto, obtenido a través de la ecuación

#### 4.8.1. Objetivo

1. Examinar si una variación de la decodificación y función fitness que asigna como coste la ponderación de dos histogramas LBP, con parámetros R=1, P=8 y R=2, P=12 respectivamente contribuyen a incrementar el rendimiento de la metodología en términos de precisión en la localización de la solución.
2. Comprobar el comportamiento de la incorporación de la heurística en la decodificación para diferentes tipos de template según su homogeneidad

#### 4.8.2. Configuración del experimento

Para este experimento se realizan 27 ejecuciones distribuidas del siguiente modo

- 9 ejecuciones para el template circuito
- 9 ejecuciones para el template pilas
- 9 ejecuciones para el template rueda

Se baraja la hipótesis de que la ampliación del radio y el vecindario podrían contribuir a aumentar la eficacia encontrando la solución para aquellos templates que exhiban regiones con texturas uniformes.

En esta ocasión, se recurre a los parámetros R (radio) y P (tamaño del vecindario) internos del operador LBP. La estrategia consiste en asignar el coste del individuo en base a dos histogramas LBP, con diferente combinación de R y P cada uno de ellos.

#### Imagen escena

La siguiente imagen es la escena original del experimento. Para el experimento, la imagen es convertida de RGB a escala de grises, como en Ilustración 4.8.2-1.

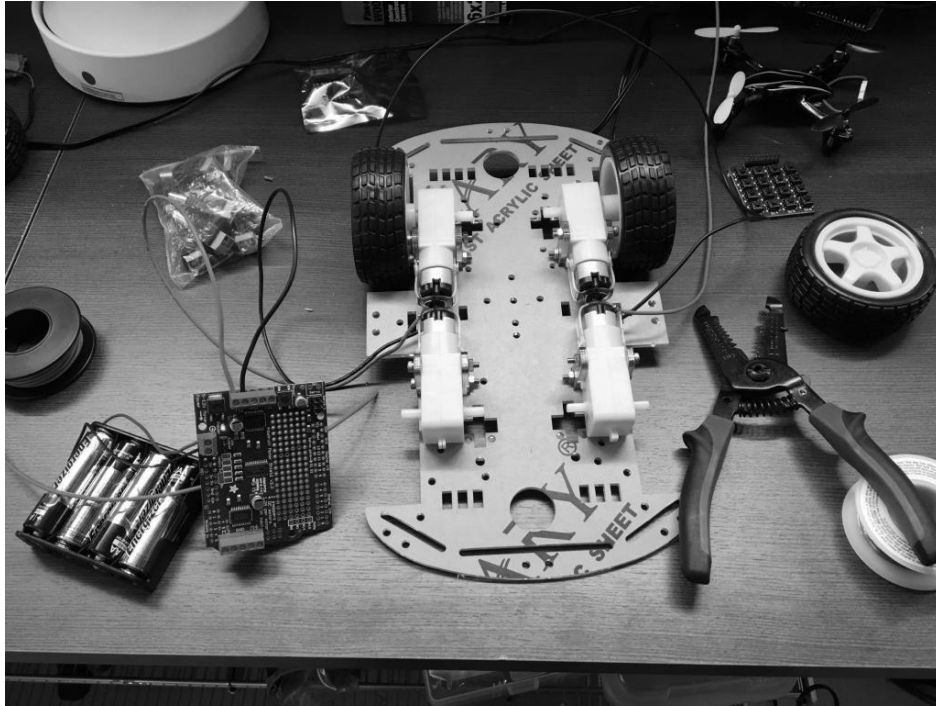


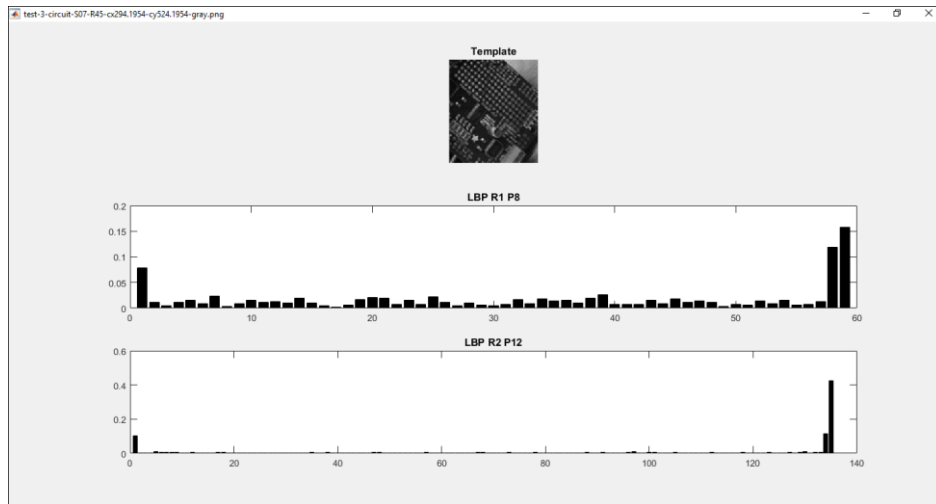
Ilustración 4.8.2-1: Escena experimento 7

## **Templates**

Todos los templates del experimento son imágenes extraídas de la escena original, rotadas 45°, escaladas por un factor de escalado 0.7 y convertidas a niveles de gris. En dicha tabla se muestran los templates con sus tamaños de acuerdo al escalado indicado anteriormente. A continuación se muestran dichos templates y cada uno de los histogramas de características LBP para cada combinación de para parámetros R y P del operador LBP

### **Template "circuito"**

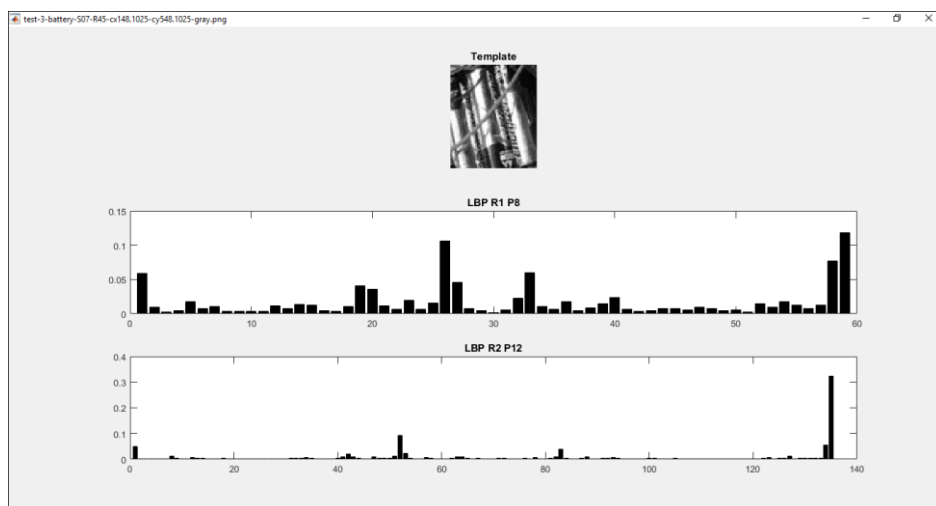
En la Ilustración 4.8.2-2 se comparan los distintos histogramas LBP para el template circuito, donde cada LBP corresponde a las combinaciones R=1 y P=8 (centro) y R=2, P=12 (abajo).



**Ilustración 4.8.2-2: Vista template circuito con diferentes histogramas LBP**

### **Template "pilas"**

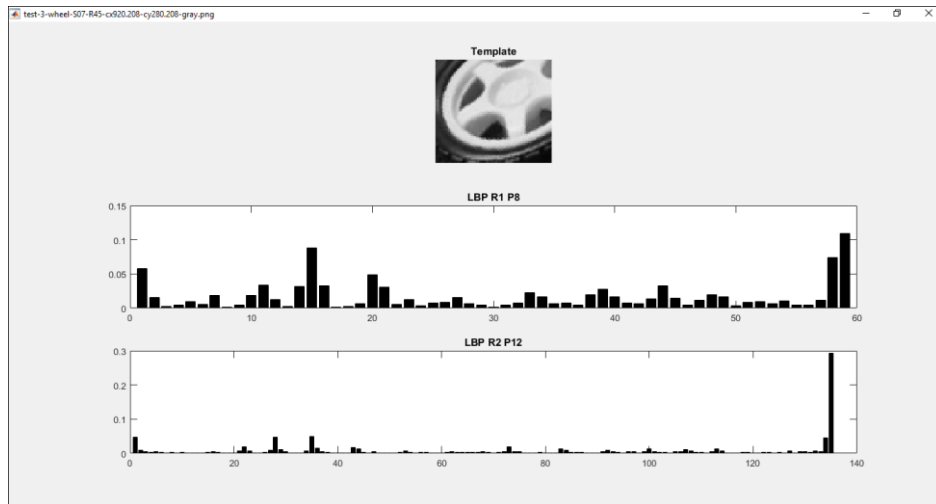
En la Ilustración 4.8.2-3 se comparan los distintos histogramas LBP para el template pilas, donde cada LBP corresponde a las combinaciones  $R=1$  y  $P=8$  (centro) y  $R=2$ ,  $P=12$  (abajo).



**Ilustración 4.8.2-3: Vista template pilas con diferentes histogramas LBP**

### **Template "rueda"**

En la Ilustración 4.8.2-4 se comparan los distintos histogramas LBP para el template rueda, donde cada LBP corresponde a las combinaciones  $R=1$  y  $P=8$  (centro) y  $R=2$ ,  $P=12$  (abajo).



**Ilustración 4.8.2-4: Vista template rueda con diferentes histogramas LBP**

En general, se observa la introducción de numerosas dimensiones en descriptor LBP al aumentar el vecindario y el radio pero dado que se está utilizando el LBP uniforme, la mayor parte de las ocurrencias siguen perteneciendo a los patrones no uniformes.

### 4.8.3. Análisis de los resultados

La presentación de los análisis sigue el siguiente orden. Primero se analizan los resultados de cada template por separado, posteriormente se valora de forma global la aplicación de este mecanismo de decodificación y asignación de coste.

El template "circuito" presenta en sus resultados, Tabla A.7-1: Resultados combinación de diferentes Operadores LBP, circuito; una tasa de éxito absoluta. Los MBF de cada ejecución se sitúan en el intervalo  $[0.0087, 0.011]$  con desviaciones de entre  $10^{-3}$  y  $10^{-4}$  en orden de magnitud. La duración media del run son aproximadamente 75 segundos. Las distancias intercentrales son todas inferiores a 11 píxeles, los genotipos presentan bastante uniformidad en las dimensiones de las coordenadas de extracción y el ángulo de rotación, dicho ángulo se desvía como máximo  $2^\circ$  de los  $45^\circ$ . El factor de escalado se ubica entre los valores 1.40 y 1.44, por lo que es bastante cercano a los 1.45 que sería la solución. Los mejores fitness se localizan en el intervalo  $[0.007, 0.01]$ .

En las ejecuciones del template "pilas" Tabla A.7-2: Resultados combinación de diferentes Operadores LBP, pilas, hallamos tasas de éxito comprendidas entre el 40% y el 90%. Los tiempos de los runs de media se sitúan cerca del minuto. Los MBF oscilan entre los valores 0.014 y 0.036; con desviaciones típicas de centésimas o milésimas. Se observan tres distancias intercentrales menores a un píxel, y una de 55 píxeles, las demás son inferiores a 16 píxeles. Los mejores genotipos, son bastante uniformes en todas las dimensiones, salvo el correspondiente al caso de la combinación 0.5, 0.8 de F y CR respectivamente donde se observa una desviación considerable en la coordenada x. Los

mejores fitness varían desde 0.009, hasta el máximo de 0.24. De nuevo ese máximo corresponde a la combinación mencionada de F y CR.

Para el template de la rueda, Tabla A.8-3: Resultados selección de un subconjunto de bins del histograma LBP, rueda, la tasa de éxito tiene el mismo mínimo y máximo que el anterior template, es decir, 40% y 90%. En cuanto a la duración de los runs, estos no superan los 350 segundos. Los MBF varían entre 0.021 y 0.039, con desviaciones estándar de  $10^{-3}$  y  $10^{-4}$  respectivamente. Todas las distancias intercentrales son inferiores a 44 píxeles, por lo que no hay candidatos que puedan suponer un óptimo local muy desviado de la posición del template original en la escena. Los genotipos presentan algo más de variabilidad que en el template anterior, observándose intervalos más amplios entre las dimensiones de los mismos. Los mejores fitness se localizan en el intervalo [0.013, 0.027].

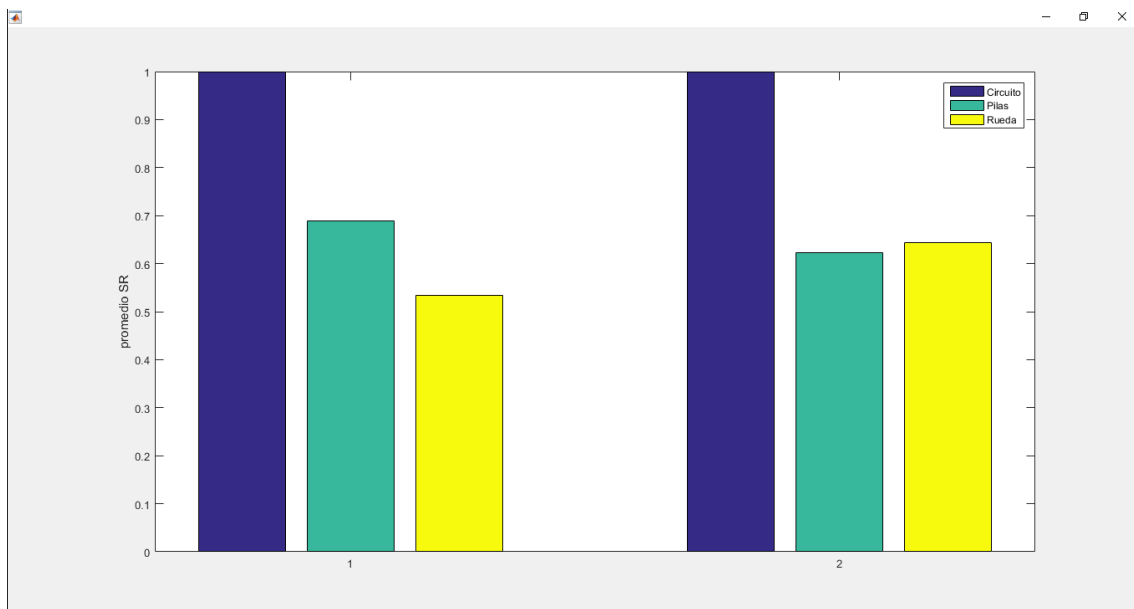


Figura 4.8.3-1: Comparación SR promediado con función que combina diferentes LBP

En la Figura 4.8.3-1 podemos comparar el SR promediado para cada template obtenido para la función *fitness* original (bloque 1) respecto a esta función *fitness* que combina diferentes Operadores LBP (bloque 2).

En general los resultados observados para este experimento no indican un empeoramiento del comportamiento del método, pero tampoco se aprecia una mejora significativa. Los tiempos de duración de los runs aumentan unos segundos como cabía esperar dado que es preciso calcular dos histogramas en lugar de uno; además el segundo de ellos analiza más vecindario y el vector de características está constituido por más hipotéticos patrones LBP.

## 4.9. Experimento 8: Selección de un subconjunto de bins del histograma LBP

Dado que en el descriptor LBP empleado, el uniforme, los patrones no uniformes tienden a ser la clase más numerosa, en este experimento se plantea introducir en la función fitness un criterio para evitar patrones con alta frecuencia (entre los que quedaría el de los patrones no uniformes).

Para ello se sustituye la función de evaluación o fitness de acuerdo a Ecuación 4.8.3-1:

$$f = \frac{\text{dist}(\text{hist}'(\text{hist}(\text{LBP}(t))) - \text{hist}'(\text{hist}(\text{LBP}(i))))}{r}$$

**Ecuación 4.8.3-1: Función Fitness que aplica un reajuste al descriptor LBP**

Donde:

- $t$  es la matriz imagen (escala de grises) del template
- $i$  es la matriz imagen (escala de grises) del individuo
- $\text{dist}$  es la distancia Euclídea, que en este caso corresponde entre el vector de características LBP del template y el vector de características LBP del individuo.
- $\text{LBP}$  es el operador LBP
- $\text{hist}$  referencia el vector de características resultante de la aplicación del operador LBP
- $r$  es el resto, obtenido a través de la siguiente ecuación:
- $\text{hist}'$  es una función que transforma el vector de características resultante tras la aplicación del operador LBP que sigue la Ecuación 4.8.3-2:

$$\text{hist}' = (\max(\text{hist}) - \text{hist}) \iff \lambda * \max(\text{hist}) < (\max(\text{hist}) - \text{hist})$$

**Ecuación 4.8.3-2: Función de reajuste para descriptor LBP**

Donde:

- $\max(\text{hist})$  es el valor máximo del histograma/vector de características obtenido tras la aplicación del operador LBP
- $(\max(\text{hist}) - \text{hist})$  define un nuevo vector
- $\lambda$  es un parámetro a modo de umbral, ajustado al valor 0.5

### 4.9.1. Objetivo

1. Examinar si una variación de la decodificación y función fitness que en lugar asignar coste mediante una comparación directa de las distancias de los

histogramas LBP template vs individuo, realiza un ajuste de dichos histogramas antes de compararlos contribuye a una mejora del rendimiento de la metodología.

2. Comprobar el comportamiento de la incorporación de la heurística para distintos templates y combinaciones de parámetros DE.

La función fitness encargada de realizar el ajuste del histograma LBP, está descrita en la sección 3.2.3 **¡Error! No se encuentra el origen de la referencia.**

#### 4.9.2. Configuración del experimento

El experimento consta de 27 ejecuciones estructuradas según los siguientes puntos:

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

#### Imagen escena

La siguiente imagen es la escena original del experimento. Para el experimento, la imagen es convertida de RGB a escala de grises, plasmada en la Ilustración 4.9.2-1.

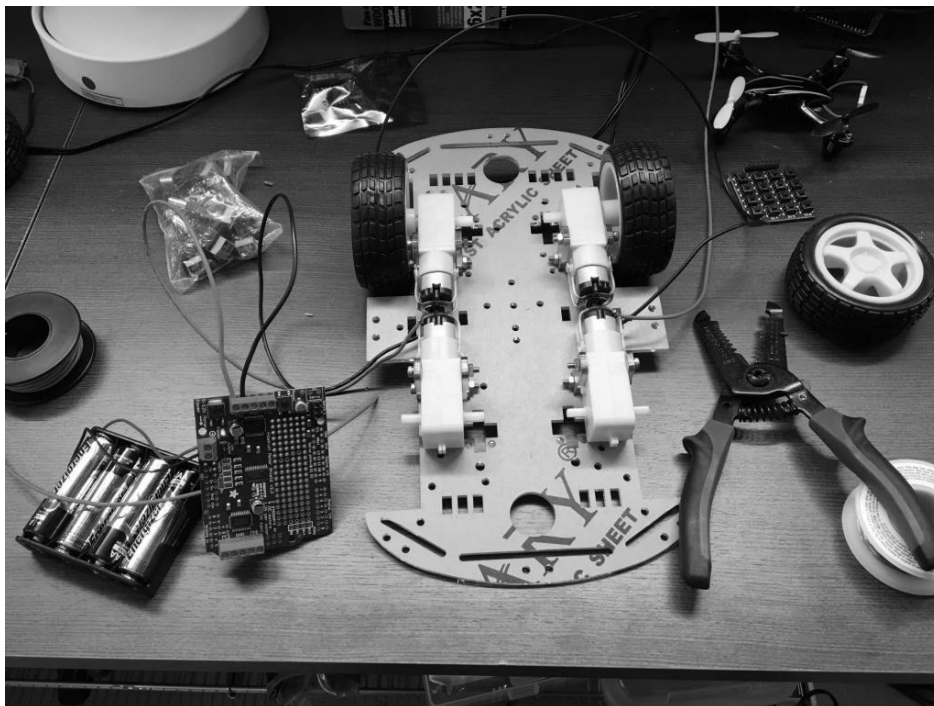


Ilustración 4.9.2-1: Escena experimento 8

## Templates

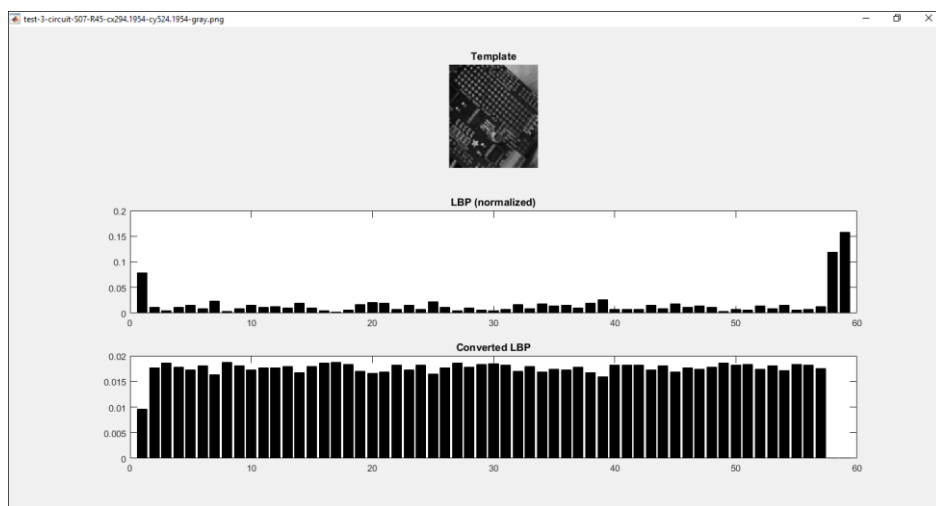
Todos los templates de este experimento se orientan al tipo de individuo "TLFSR" a los cuales les han sido aplicadas las mismas transformaciones geométricas tanto para el cambio de escala (0.7) como para la rotación (45°), tras extraerlos de la imagen y convertirlos a imágenes en niveles de gris.

A continuación, se muestran los distintos templates, junto con la visualización de su histograma LBP normalizado y el histograma LBP normalizado una vez aplicada la conversión. La lógica de la conversión puede consultarse en la sección de Funciones Fitness.

### Template "circuito"

En la Ilustración 4.9.2-2, se puede observar que tras aplicar el criterio descrito por la Ecuación 4.8.3-2 ha conllevado a deshacerse de la clase de patrones no uniformes, (bin 59) pero también se ha perdido el bin 58 que describe algún patrón que si es uniforme.

No obstante, se ve como han sido realizados todos los demás patrones con ocurrencias para este template.

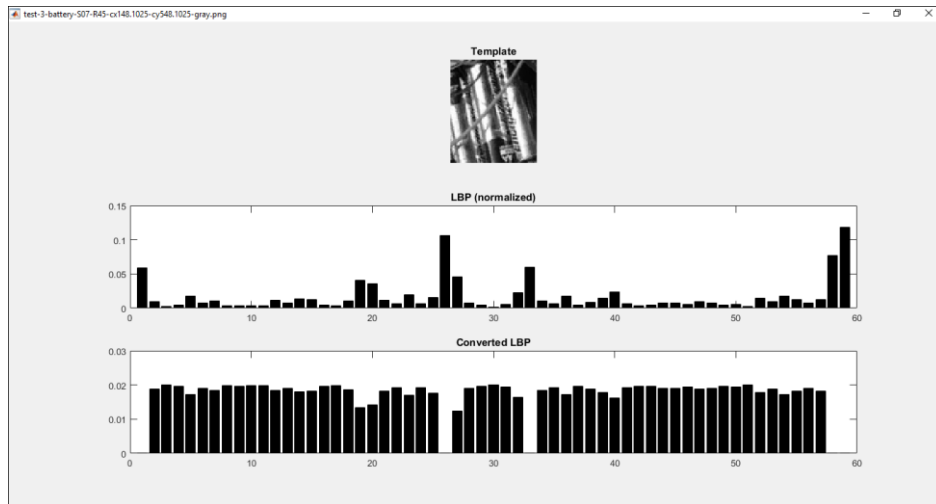


**Ilustración 4.9.2-2: Vista template circuito con histograma LBP ajustado**

### Template "pilas"

En la Ilustración 4.9.2-3, se puede observar que tras aplicar el criterio descrito por la Ecuación 4.8.3-2, se pierden 4 bins, el de los patrones no uniformes y otros tres adicionales que tal vez contribuyesen a una mejor descripción del template en términos LBP.

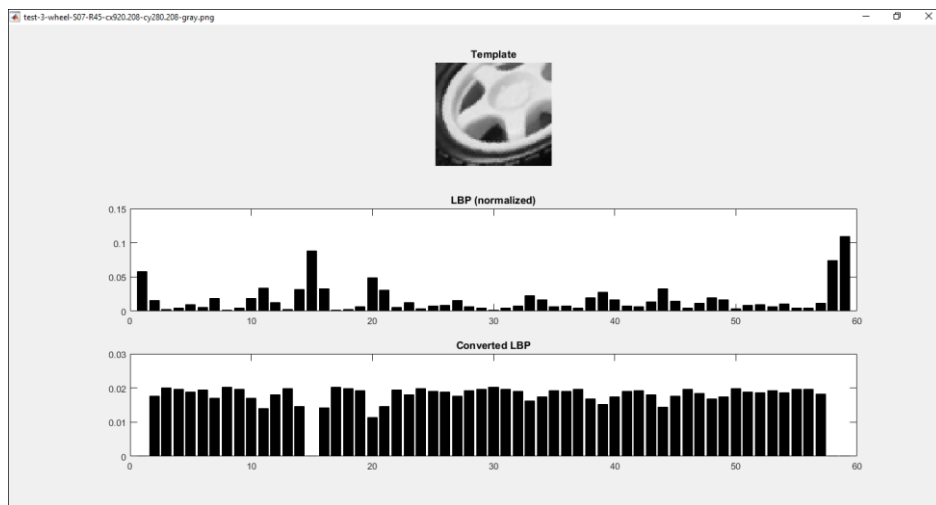




**Ilustración 4.9.2-3: Vista template pilas con histograma LBP ajustado**

### Template "rueda"

En la Ilustración 4.9.2-4, se puede observar que tras aplicar el criterio descrito por la Ecuación 4.8.3-2, se pierden 3 bins tras la transformación, donde está incluido el de patrones no uniformes.



**Ilustración 4.9.2-4: Vista template rueda con histograma LBP ajustado**

En lugar de simplemente no considerar el bin de los patrones no uniformes y recurrir a esta aproximación, tiene como objeto únicamente verificar si considerar sólo patrones uniformes poco frecuentes dado un template, contribuye a su correcta identificación en el proceso de *Pattern Matching*.

### 4.9.3. Análisis de los resultados

El estudio de los resultados se plasma del siguiente modo. Primero se exponen los resultados hallados para cada template por separado, después se valora el rendimiento global de la metodología tras aplicar esta heurística.

Los resultados para el template del circuito, en cuanto a tasa de éxito son bastante satisfactorios, Tabla A.8-1: Resultados selección de un subconjunto de bins del histograma LBP, circuito, todas las ejecuciones salvo una alcanzan el 100% de éxito. Además, el MBF de dichas ejecuciones se sitúa en órdenes de magnitud de  $10^{-3}$  y  $10^{-4}$  para todas ellas, con desviaciones estándar que llegan incluso a  $10^{-5}$ . El tiempo total de ejecución por run es de unos 45 segundos, por lo que esta heurística no aumenta el tiempo de ejecución respecto a la decodificación básica.

Las distancias intercentrales son inferiores a 9 píxeles en todos los casos, siendo en una ejecución inferior a un píxel. Los genotipos de la solución con menor fitness de cada run presentan homogeneidad en todas sus dimensiones, estando la coordenada x cercana al valor 200, la coordenada y cercana a 430, el factor de escalado entre 1.40 y 1.44, así como el ángulo de rotación es cercano a  $45^\circ$ . Los mejores fitness se observan pertenecientes al intervalo  $[0.0007, 0.001]$ .

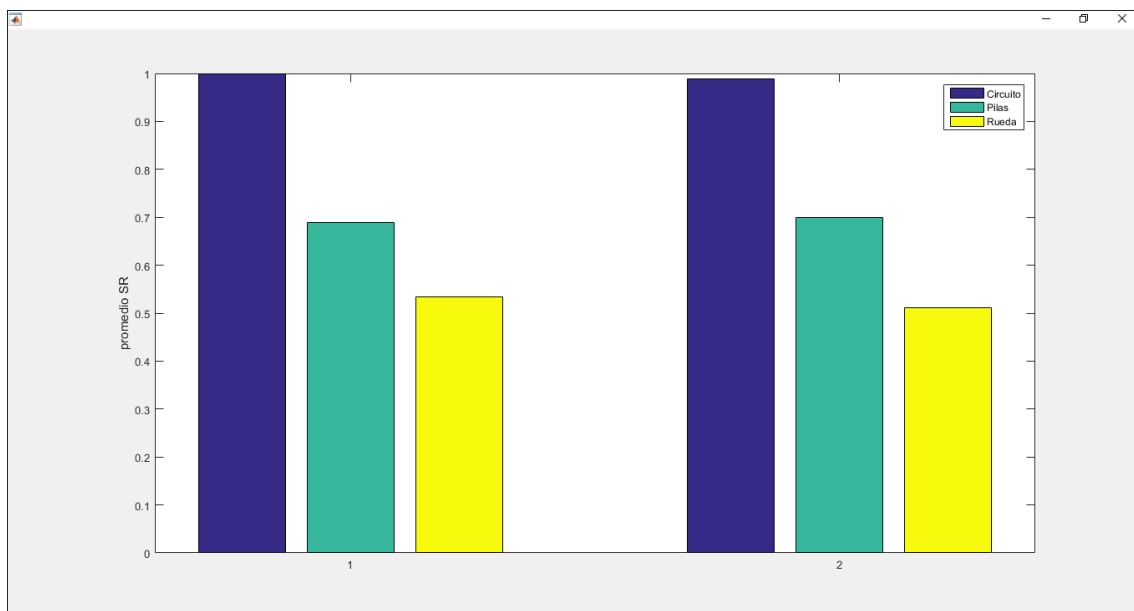
Por su parte, Tabla A.8-2: Resultados selección de un subconjunto de bins del histograma LBP, pilas, el template de las pilas solo alcanza la tasa de éxito total en dos ejecuciones, las correspondientes a la combinación de parámetros F y CR 0.5 ambos, y F=0.8 y CR=0.5. Las demás ejecuciones varían en su SR del valor 0.3 al valor 0.8. El MBF de este conjunto de ejecuciones se observa en el intervalo  $[0.0032, 0.0071]$  con desviaciones estándar del orden de las milésimas. La duración media de los runs es de aproximadamente 40 segundos.

Las distancias intercentrales, presentan algo más de varianza, cinco de ellas están por debajo de los 9 píxeles, de las restantes, una incluso supera los 50 píxeles. Sobre la similitud de las dimensiones de los genotipos encontramos que la coordenada x parece distribuirse en algunos casos cercana al valor 67 y en otros cercana al valor 30; todos los valores de los genotipos en la coordenada y están entre 430 y 492 píxeles. El factor de escalado tiene a acercarse al óptimo, que sería 1.45 salvo en un caso; donde es 0.9 y además precisamente coincide con la ejecución donde la distancia intercentral supera los 50 píxeles (ejecución cuya combinación de parámetros F y CR es 0.8 ambos). El ángulo de rotación por su parte, no se desvía más de tres grados de los  $45^\circ$ . Los costes se sitúan en el intervalo  $[0.014, 0.038]$ , siendo la cota máxima el coste de la ejecución cuyos parámetros F y CR equivalen a 0.8 nuevamente.

El último template, el de la rueda, Tabla A.8-3: Resultados selección de un subconjunto de bins del histograma LBP, rueda, presenta SR de como mínimo un éxito del 30% y como máximo del 80%. El MBF está entre 0.05 y 0.06 para estas ejecuciones, con desviaciones estándar del orden de magnitud de las milésimas o  $10^{-3}$ . La duración media del run es cercana a los 33 segundos en todo momento.

Las distancias intercentrales son inferiores a 41 píxeles en todas las ejecuciones, siendo incluso menor a un píxel en una de ellas. Los genotipos presentan una variabilidad leve en sus dimensiones, situándose los costes de estos genotipos en el intervalo [0.0021, 0.0042].

En la Figura 4.9.3-1 puede apreciarse el SR promediado obtenido para cada template según la función fitness original (bloque 1) respecto a esos mismos valores obtenidos por la función fitness analizada en este experimento (bloque 2).



**Figura 4.9.3-1: Comparación SR promediado de la función fitness original respecto a la que considera un subconjunto de bins del histograma**

En términos locales a este experimento, los resultados muestran que el único template que haya podido recibir algún beneficio del enfoque que aporta esta decodificación es el del circuito. Los otros dos templates no parecen haber recibido ventajas ni tampoco desventajas por emplear esta técnica para la asignación del coste.

Esto, puede reforzar la noción de que la naturaleza del template, en cuanto a características LBP, como la de la propia escena también en términos LBP, tiene una influencia especialmente relevante a la hora de localizar objetos sirviéndonos de este operador como criterio de similitud. Dicho de otro modo, encontrar un objeto cuyas características LBP se diferencien del resto de características LBP que pudieran existir en otras regiones de la escena será más sencillo, independientemente de que el template sea

uniforme o accidentado. Podría ser un problema equivalente buscar un template accidentado en una imagen que en las demás regiones tienda a ser uniforme, que buscar un template uniforme en una escena que tienda a ser abrupta.

## 4.10. Experimento 9: Búsqueda Local

Para este experimento se incorpora una etapa de búsqueda local dentro del proceso que sigue la metodología, concretamente al final de cada generación del Algoritmo Evolutivo.

Para ello, se adoptan tres posturas distintas, que comparten ciertas similitudes.

La primera postura, consiste en una búsqueda local simple, la denominamos “fixed single” porque, para el mejor individuo de cada generación se inicia una búsqueda local donde dado el vector genotipo de este mejor individuo se crean siguiendo una función gaussiana, 10 individuos vecinos.

Para la obtención de los cromosomas  $x$  de los individuos vecinos, se emplea una función gaussiana de media  $x$  y de desviación estándar 2. Lo mismo aplica al cromosoma  $y$ .

Para los cromosomas  $\alpha$ , la función gaussiana tiene de media  $\alpha$  y de desviación estándar  $10^{-2}$ ; para los cromosomas  $\theta$ , la media es  $\theta$  y la desviación estándar es 0.5.

Para más información sobre cómo generar una distribución gaussiana, consultar (Mathworks, 2017).

La segunda aproximación “threshold single” coincide en la primera en que para cada generación se elige al mejor individuo para la búsqueda local y también coincide en cómo generar los vecinos y los valores de cada cromosoma o dimensión que usar como media en la distribución gaussiana, así como en los valores de las desviaciones.

La diferencia, es que en esta aproximación se irán generando vecinos de manera iterativa y comprobando su fitness; en el momento en el que un vecino tuviera mejor fitness, este vecino reemplazaría en la población al miembro original. Si dado un umbral de generación de vecinos no se encuentra ninguno mejor, el proceso de búsqueda local termina, con el individuo original permaneciendo en la población.

Por último, la aproximación “fixed multi”, es exactamente igual que la primera “fixed single”, con la diferencia de que en lugar de buscar vecinos para el mejor individuo de la generación en cuanto a fitness, se buscan vecinos para los 10 mejores individuos.

### 4.10.1. Objetivo

Los objetivos esenciales de este experimento son:

1. Observar el rendimiento de la metodología incluyendo una fase de búsqueda local, basada en tres estrategias diferentes.
2. Comprobar si la incorporación de dichas estrategias de búsqueda local influyen en la eficacia de los distintos parámetros F y CR motor principal de búsqueda

#### **4.10.2. Configuración del Experimento**

El experimento consta de 81 ejecuciones agrupadas del siguiente modo:

Para la búsqueda local "fixed single"

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

Para la búsqueda local "threshold single"

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

Para la búsqueda local "fixed multi"

- 9 ejecuciones para el template "circuito", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "pilas", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}
- 9 ejecuciones para el template "rueda", cada ejecución corresponde a una combinación de pares F y CR del conjunto {0.2, 0.5, 0.8}

#### **Escena y templates**

La escena y los templates pueden consultarse en la Ilustración 4.2.2-1 y en Tabla 4.2.2-1, es decir son las mismas que en el primer experimento.

#### **4.10.3. Análisis de los resultados**

Los resultados hallados se exponen del siguiente modo. Primero, para cada par tipo de búsqueda - template, se analizan los resultados de manera independiente. A continuación

se valoran de forma contrastada los resultados generales entre cada tipo de búsqueda. Por último se reflexiona sobre los beneficios y desventajas de introducir una fase de búsqueda local en la metodología.

Para la búsqueda local "fixed single" y el template "circuito", Tabla A.9-1: Resultados búsqueda local fixed single, circuito, los resultados de SR son del 100% en todos los casos salvo para la combinación  $F=0.2$  y  $CR=0.5$ , donde es del 90%. Las distancias intercentrales son todas inferiores a 6 píxeles. El MBF de cada ejecución siempre es un valor comprendido en el intervalo  $[0.007, 0.010]$  con desviaciones estándar entre  $10^{-3}$  y  $10^{-4}$ . El tiempo de ejecución por run es de un minuto. Los mejores genotipos exhiben valores bastante cercanos entre sí en todas sus dimensiones, con costes alrededor del valor 0.007.

En el caso "fixed single" con template "pilas", Tabla A.9-2: Resultados búsqueda local fixed single, pilas, el SR alcanza el 100% en una ejecución, las demás varían de 0.7 a 0.9 salvo una, que siendo 0.4 corresponde a la combinación  $F$  y  $CR$  equivalentes a 0.8. Las distancias intercentrales, salvo para este caso mencionado, son menores a 12 píxeles, siendo dos de ellas inferiores a un píxel. El MBF se sitúa en el intervalo  $[0.01, 0.03]$  con desviaciones esta vez entre  $10^{-2}$  y  $10^{-4}$ . El tiempo medio de duración del run es de unos 50 segundos. Los mejores genotipos, de nuevo presentan valores similares entre ellos, salvo por una desviación de la coordenada  $x$  en uno de ellos. Sus fitness, oscilan entre 0.008 y 0.01.

El template "rueda" incluyendo una búsqueda de tipo "fixed single", Tabla A.9-3: Resultados búsqueda local fixed single, rueda, obtiene tasas de éxito de entre el 30% y el 70%. En la mayor parte de los casos las distancias intercentrales son inferiores a 5 píxeles, llegando un caso incluso a ser inferior a un píxel. El MBF se sitúa entre  $[0.02$  y  $0.03]$ , con desviaciones estándar del orden de magnitud de las centésimas y las milésimas. El tiempo de duración medio del run se sitúa en los 40 segundos. Los genotipos, presentan mayor varianza en las coordenadas  $x$  e  $y$  que en las del factor de escalado y rotación, sus fitness, se comprenden en el intervalo  $[0.11, 0.25]$ .

En Tabla A.9-4: Resultados búsqueda local threshold single, circuito, "threshold single" para el template "circuito" aporta un SR absoluto en todas las ejecuciones. Todas las distancias intercentrales son inferiores a 6 píxeles. El MBF se sitúa entre 0.007 y 0.009, con desviaciones de órdenes  $10^{-4}$  y  $10^{-5}$  de magnitud. El tiempo de duración medio por run aumenta respecto al método de búsqueda anterior hasta los 95 segundos aproximadamente. Los mejores genotipos presentan dimensiones de factor de escalado y rotación muy cercanas. Las coordenadas  $x$  e  $y$  de nuevo son más variables. Los fitness se sitúan en el intervalo  $[0.0067, 0.0093]$ .

EL template "pilas" mejora en sus resultados, Tabla A.9-5: Resultados búsqueda local threshold single, pilas, obteniendo dos ejecuciones con una tasa de éxito del 100%. Las demás varían del 40% al 90%. Las distancias intercentrales, en esta ocasión están todas por debajo de los 8 píxeles, siendo una de ellas inferior a un píxel. El MBF varía en las ejecuciones dentro del intervalo [0.02, 0.01], con desviaciones estándar de centésimas y milésimas. El tiempo medio de duración de los runs presenta una mayor variabilidad. La media se situaría en torno a los 74 segundos. Una vez más, las principales dimensiones beneficiadas por el mecanismo de búsqueda local corresponden al cambio de escala y la rotación. Las coordenadas x e y, pese a pertenecer a intervalos más estrechos, siguen presentando menos homogeneidad. Los fitness se ubican en el intervalo [0.008, 0.01]

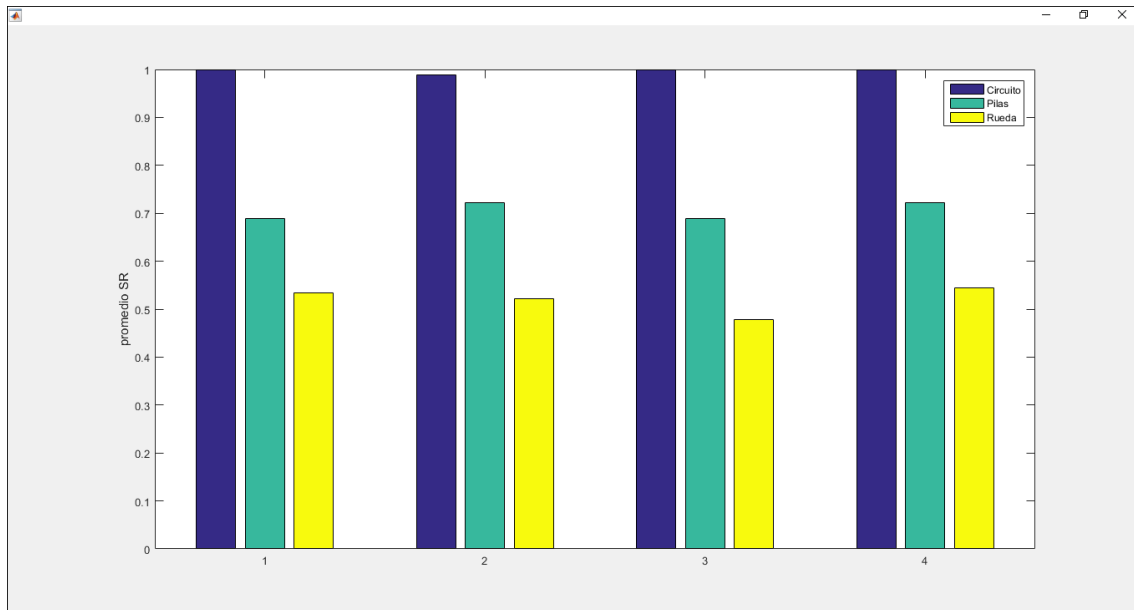
El template "rueda" no alcanza, para el tipo de búsqueda local "threshold single" Tabla A.9-6: Resultados búsqueda local threshold single, rueda el éxito absoluto en ninguna ejecución, pero solo una de ellas es inferior al 50%, siendo del 40%. Las distancias intercentrales, salvo para dos ejecuciones, inferiores a 4 píxeles. El tiempo de duración medio del run estaría en unos 60 segundos. Los mejores genotipos presentan algo menos de homogeneidad en sus dimensiones que los casos anteriores, con fitness en un intervalo más elevado y ancho respecto a los otros dos templates; [0.011, 0.024]

El último mecanismo de búsqueda local probado, "fixed multi", mantiene la tasa de éxito del 100% para el template "circuito", Tabla A.9-7: Resultados búsqueda local fixed multi, circuito, donde las distancias intercentrales siguen siendo inferiores a 6 píxeles. El MBF se sitúa entre los valores 0.077 y 0.01, con desviaciones comprendidas en órdenes de magnitud  $10^{-3}$  y  $10^{-4}$ . El tiempo medio de duración del run se dispara a los 160 segundos de media, lo cual supone un ascenso considerable. Los mejores genotipos son visiblemente homogéneos y sus fitness se dispersan por el intervalo [0.0062, 0.0089].

Para esta búsqueda local, el template "pilas" Tabla A.9-8: Resultados búsqueda local fixed multi, pilas, supera el 50% de SR en todas las ejecuciones, con distancias intercentrales menores a 8 píxeles salvo en una ejecución donde son 27. El MBF mínimo es 0.012 y el máximo 0.028. Los órdenes de magnitud de las desviaciones estándar de los MBF son de  $10^{-2}$  y  $10^{-3}$ . La duración media del run se aproxima a los 140 segundos. Los intervalos de ocurrencia de las dimensiones de los genotipos son más anchos, ubicándose los fitness entre los valores 0.0085 y 0.016.

En el caso del template "rueda", Tabla A.9-9: Resultados búsqueda local fixed multi, rueda, la tasa de éxito de las ejecuciones es de entre un 30% y 70%, siendo las distancias intercentrales en todas las ejecuciones menos una, inferiores a 4 píxeles. Los MBF presentan valores entre 0.029 y 0.034 con desviaciones estándar generalmente de magnitud  $10^{-2}$ . Los mejores genotipos encontrados presentan bastante homogeneidad,

pero la dimensión del factor de escalado solo asciende en uno de ellos hasta alcanzar el valor 1.41. Los fitness, se localizan en el intervalo [0.011, 0.021].



**Figura 4.10.3-1: Comparación SR promediado estrategias de búsqueda local**

En la Figura 4.10.3-1 se comparan los SR promediados de cada uno de los templates en las siguientes situaciones: el bloque 1 corresponde a los valores obtenidos para cada template en los experimentos sin búsqueda local para la función *fitness* original; el bloque 2 corresponde a los valores SR promediados para la estrategia “fixed single”, el tercero para la estrategia “threshold single” y el último bloque para la estrategia “fixed multi”.

En ningún caso se aprecia una mejora en el SR promediado para ningún template, además no hay que olvidar que la introducción de la fase de búsqueda local es costosa en computacionalmente hablando en todos los casos, especialmente para la estrategia “fixed multi”.





## **CAPÍTULO 5. USO DEL MÉTODO EN UN PROBLEMA REAL**

---



## 5.1. Consideraciones previas

La retina de los vertebrados es un tejido sensible a la luz situado en la superficie interior del ojo. Es similar a una tela donde se proyectan las imágenes. La luz que incide en la retina desencadena una serie de fenómenos químicos y eléctricos que finalmente se traducen en impulsos nerviosos que son enviados hacia el cerebro por el nervio óptico.

La retinografía es una técnica que se utiliza en medicina para obtener fotos a color de la retina, es decir imágenes de retina. La retina es la capa de tejido sensible a la luz que se encuentra en el interior del ojo, gracias a la cual es posible la visión.

Las imágenes escena de este experimento han sido tomadas del conjunto de imágenes de Retina disponibles en el conjunto DIARETDB1, *Standard Diabetic Retinopathy Database Calibration 1*.

Se trata de una base de datos pública para el benchmarking de métodos de detección automáticos de retinopatía diabética en imágenes digitales. El propósito del diseño del conjunto de imágenes es fijar un dataset y protocolo de pruebas que pueda ser usado indistintamente por diferentes sistemas de detección.

El conjunto de imágenes consiste en 89 imágenes de retina en RGB de las cuales 84 contienen signos de microaneurismas y 5 son consideradas normales de acuerdo al criterio de los expertos que participaron en la evaluación. Las imágenes fueron capturadas empleando la misma cámara digital con configuraciones diferentes. Los datos de estas configuraciones corresponden a situaciones prácticas habituales, con el objetivo de proveer un cierto margen que permita evaluar el rendimiento de los resultados de diagnóstico automático en condiciones normales.

Para la obtención de valores de Ground Truth, fueron recogidas las evaluaciones de cuatro expertos a los que se les facilitó una misma herramienta software para realizar sus evaluaciones. Se considera experto a un médico oftalmólogo con una sólida experiencia en el diagnóstico de retinopatías. Las instrucciones entregadas a estos expertos consistían en una serie de indicaciones relacionadas con señalar las áreas donde observasen microaneurismas, hemorragias y exudados; debían incluir en la señalización cualquier píxel que considerasen perteneciente a cualquiera de los casos mencionados.

Adicionalmente, se indicaba que incluyesen un valor de confianza probabilístico respecto a su propia anotación, dentro del set {<50%, >50%, 100% }.

Los datos recopilados al margen de las instrucciones por cada uno de los expertos manifestaron diferencias considerables, por lo que tales datos fueron fusionados con el fin de aportar una mayor precisión a los valores de Ground Truth.

## 5.2. Objetivo

- Aplicación del método implementado para la detección del Disco Óptico en la base de imágenes Dataset DiaretDB1

## 5.3. Configuración del Experimento

Para este experimento se realiza para cada una de las imágenes de la base de imágenes una ejecución, donde:

- El modelo de individuo es:  $[x, y, \alpha]$ , por lo que la decodificación considera cambio de escala.
- El número de runs es 10
- La combinación de parámetros DE, F y CR es 0.8 y 0.5 respectivamente
- No se aplica búsqueda local
- Le es aplicado un cambio de escala inicial a las escenas de factor 0.2
- Le es aplicado a las coordenadas de Ground Truth el mismo factor de escaldado 0.2
- Las imágenes son convertidas a niveles de gris

En total son 89 ejecuciones, una para cada imagen de retina, con 10 runs cada ejecución. Es importante no perder de vista este aspecto de la configuración para entender los resultados.

### 5.3.1. Template

Para implementar este experimento, ha sido necesario constituir un vector LBP “sintético” a partir del conjunto de todos los Discos Ópticos de las imágenes de retina de DIARETDB1.

Solo se dispone de los valores de Ground Truth del centro del disco óptico, proporcionado en base al criterio de los expertos.

Se extrajo manualmente de cada imagen de la base, las coordenadas de un rectángulo que contiene al disco óptico en su interior dejando un pequeño margen exterior al disco óptico de algunos píxeles.

Una vez obtenidas las coordenadas de cada uno de los rectángulos, se recorrió el set de imágenes para:

- Extraer una subimagen de acuerdo a las coordenadas correspondientes de cada imagen
- Calcular el histograma LBP de la subimagen
- Incorporar el descriptor a una matriz de descriptores

Una vez obtenida la matriz de los descriptores, a partir del cálculo del valor medio de cada una de sus columnas, se constituye cada una de las dimensiones del template LBP sintético.

Este mismo template LBP es utilizado para localizar en el experimento, los distintos Discos Ópticos, siguiendo la metodología para el caso de tres dimensiones, es decir incluyendo cambio de escala.

La razón por la que es preciso introducir el cambio de escala en el genotipo, es debido a que la extracción manual de las coordenadas dio lugar a subimágenes de distinto tamaño durante la fase de construcción del vector LBP sintético.

### 5.3.2. Umbral de la Tasa de Éxito

Para este experimento, si la distancia entre el centro devuelto por el método y el centro real del DO es menor o igual a un radio medio de DO, para la escala considerada dicho radio medio son 18.815 píxeles. Este valor se obtiene aplicando la Ecuación 5.3.2-1:

$$R_{DO(px)} = \alpha \sqrt{A_{(px)} / A_{(mm)}} R_{DO(mm)}$$

**Ecuación 5.3.2-1: Radio del Disco Óptico**

Donde:

- $R_{DO(px)}$  es el radio del disco óptico en píxeles, y también el SRT o umbral.
- $\alpha$  es un factor de escalado, que en este caso es 0.2
- $A_{(px)}$  es una constante de DIARETDB1, de valor 1,583,677
- $A_{(mm)}$  es una constante de DIARETDB1, de valor 153.1
- $R_{DO(mm)}$  es una constante de DIARETDB1, de valor 0.925

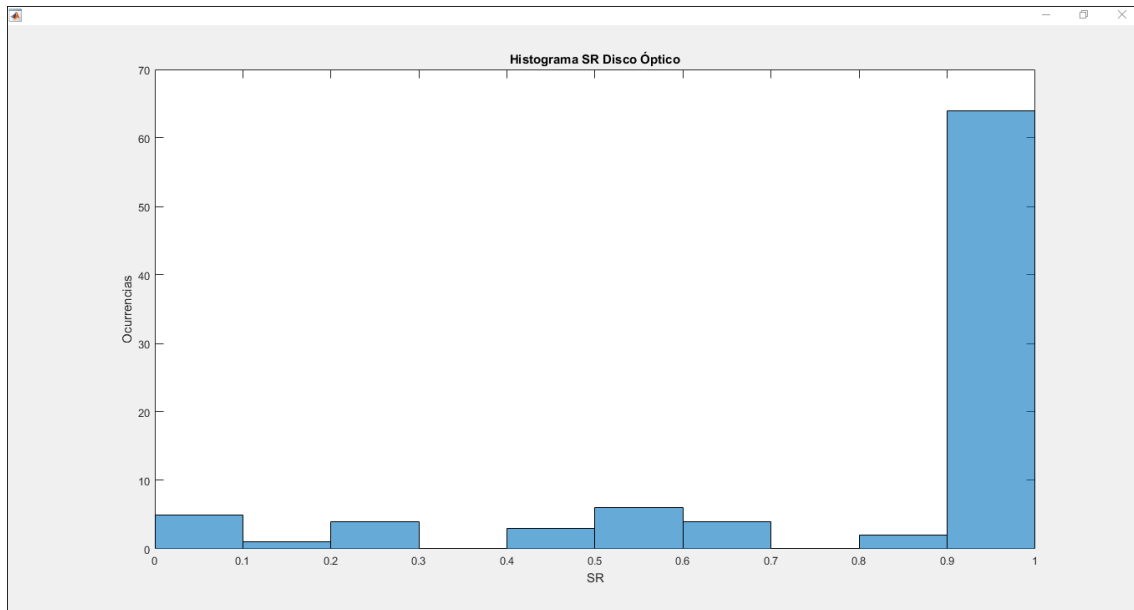
## 5.4. Análisis de los resultados

En el Anexo A, en la Tabla A.10-1: Resultados Retina Disco Óptico, están disponibles los resultados del experimento. A continuación se presentan los resultados generales del mismo.

### 5.4.1. Tasa de éxito promediada

En primer lugar, observamos los resultados para la tasa de éxito. No olvidemos que:

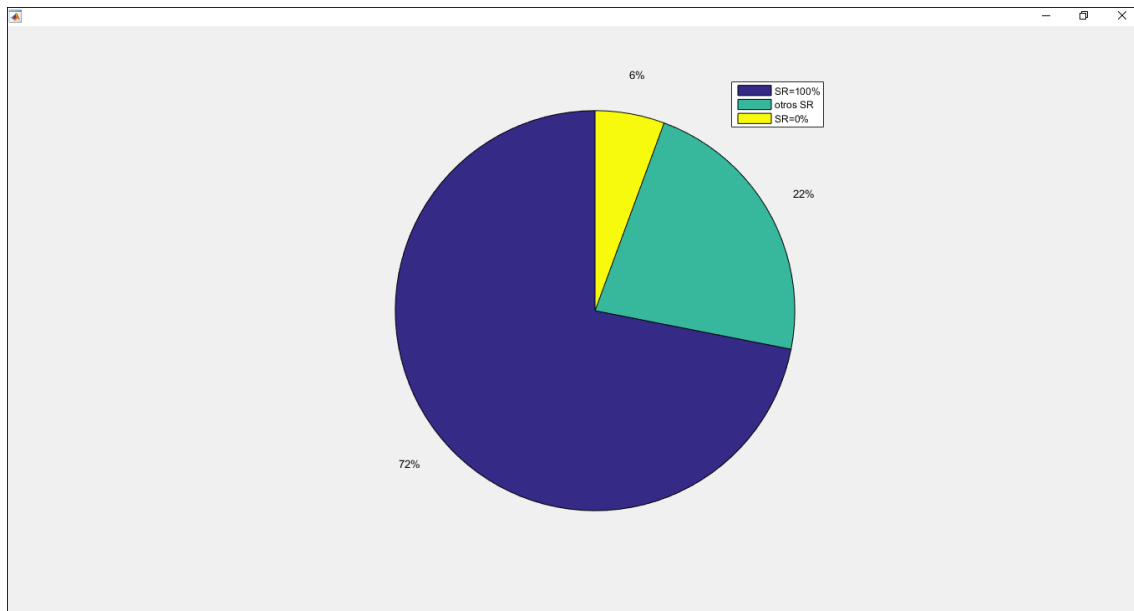
- Para cada imagen del *dataset* se hace una única ejecución con 10 runs
- Cada fila de la tabla Tabla A.10-1 corresponde a una única ejecución (de 10 runs) y a una sola imagen de retina



**Figura 5.4.1-1: Histograma DO tasa de éxito**

En la se aprecia que para una gran parte de las imágenes de retina, el método exhibe un promediado de SR superior al 90%, es decir que para la mayor parte de los runs de cada ejecución (y por tanto cada imagen de retina) se ha alcanzado un SR superior al 90%.

Concretamente, según puede observarse en laFigura 5.4.1-2, el 72% de las ejecuciones tuvieron una tasa de éxito superior al 90%, el 6% de las ejecuciones no lograron encontrar ninguna solución válida, el 22% restante encontraron soluciones en algunos de sus runs.



**Figura 5.4.1-2: Diagrama de tartas DO**

### 5.4.2. Distancias intercentrales

En la siguiente figura se representan las distancias intercentrales de la mejor solución encontrada para cada una de las imágenes respecto al Ground Truth. La media del conjunto de las mejores distancias intercentrales se sitúa en torno a los 14 píxeles.

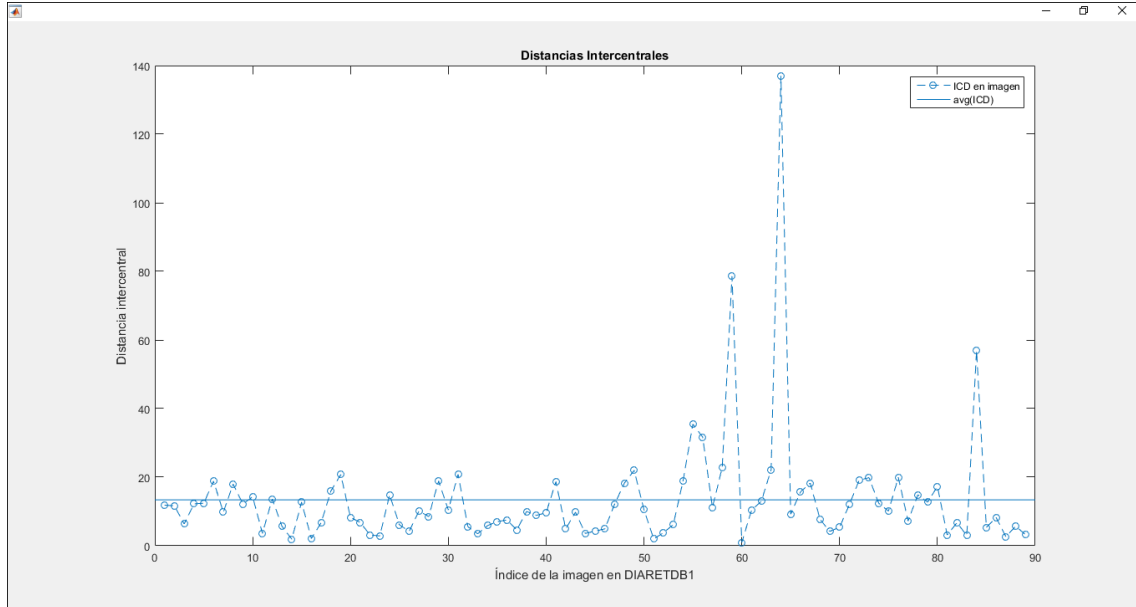


Figura 5.4.2-1: Distancias intercentrales DO

### 5.4.3. MBF

El MBF de cada ejecución correspondiente a cada imagen se representa en el siguiente gráfico, mediante una línea continua, las desviaciones estándar del MBF de cada ejecución se representan con una línea discontinua.

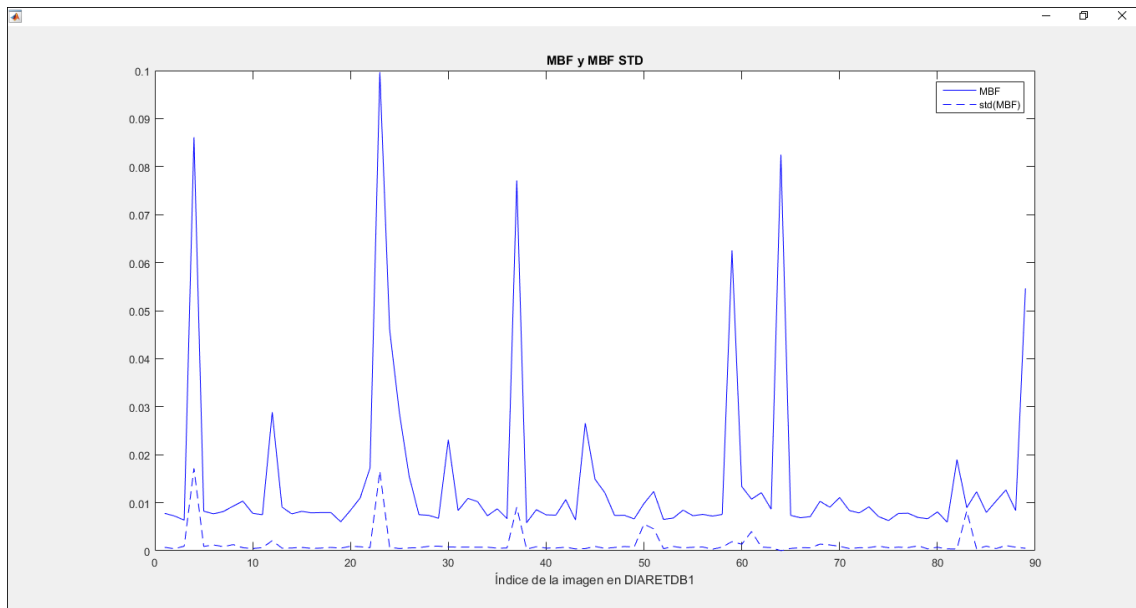


Figura 5.4.3-1: DO MBF y std (MBF)



Se puede apreciar una cierta correlación en aquellas imágenes donde el MBF aumenta considerablemente, la desviación estándar lo hace de un modo parecido.

#### 5.4.4. Duración de las ejecuciones y los runs

Como máximo la ejecución dura unos dos minutos y medio y como mínimo unos dos minutos. Esta variabilidad de la duración de la ejecución puede deberse a tratarse de un problema con cambio de escala. Al haber distintos tamaños entre las matrices de los individuos y dado que el LBP recorre dichas matrices, tardará en unos casos más que en otros.

#### 5.4.5. Mejores Fitness

El último gráfico presentado es el de los mejores *fitness* encontrados en cada imagen de retina. Destaca que para la mayor parte de las soluciones el coste de los mejores individuos es inferior a 0.01, y que para algunas imágenes el coste se dispara.

La única razón por la que ha sido incluido este gráfico es por contribuir a identificar en qué imágenes de retina no se ha logrado encontrar la solución.

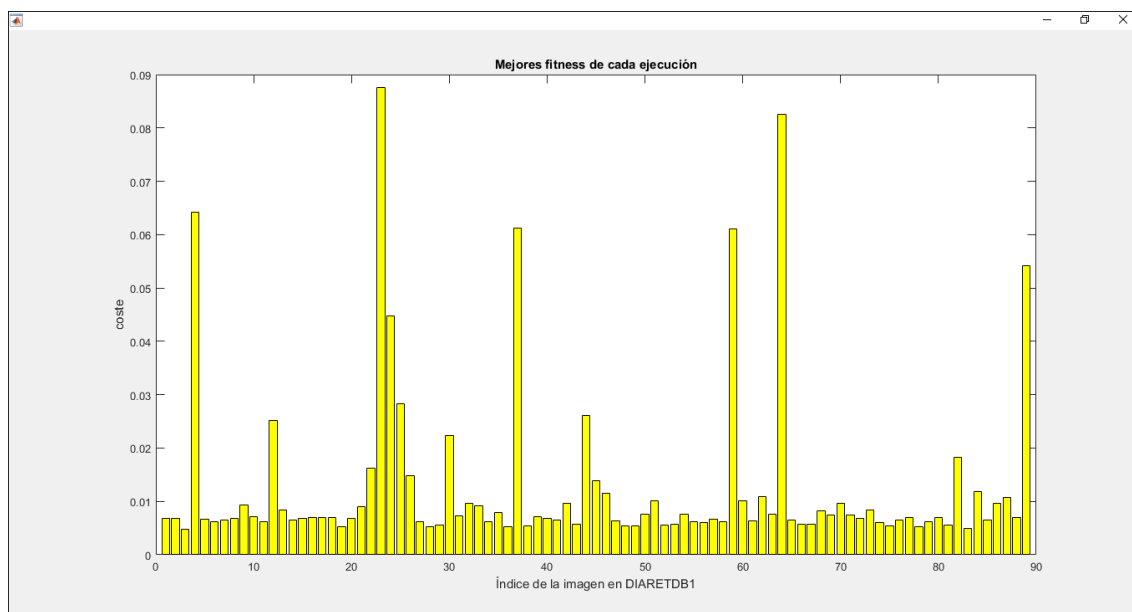
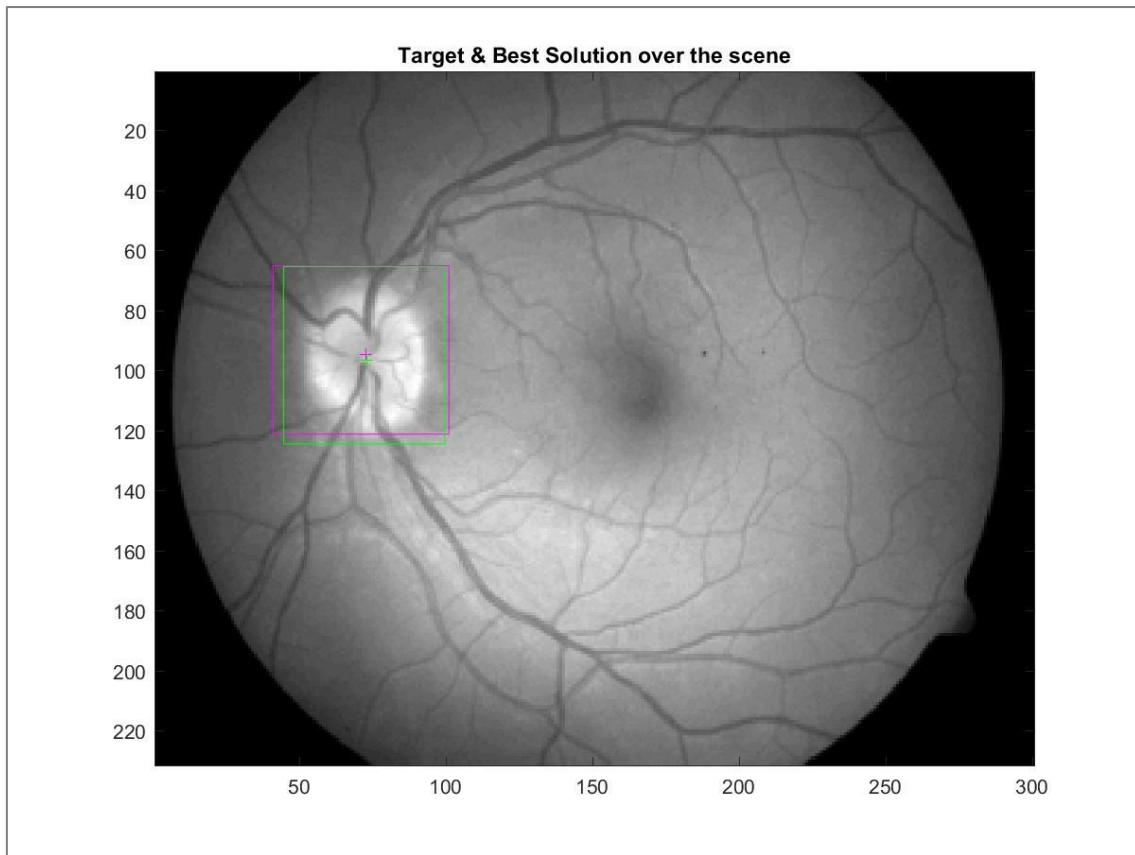


Figura 5.4.5-1: DO mejores fitness encontrados

#### 5.4.6. Resultados sobre las imágenes de retina

Atendiendo a las figuras **¡Error! No se encuentra el origen de la referencia.** y **¡Error! No se encuentra el origen de la referencia.**, es posible identificar rápidamente algunos ejemplos donde se ha encontrado con precisión el Disco Óptico en la escena así como aquellas donde el método ha sufrido dificultades. A continuación se ilustran algunos de estos casos.

En la siguiente ilustración, mostramos los resultados para la imagen de retina cuyo índice en la base de imágenes es el 51:



**Ilustración 5.4.6-1: DIARETDB1 resultado en imagen de retina, índice 51**

Tal y como puede apreciarse en la, la detección del Disco Óptico ha sido especialmente precisa en términos de cercanía del centro de la imagen correspondiente al mejor individuo encontrado (magenta) respecto a la solución global (verde).

También se observa que en este caso, el Disco Óptico queda totalmente incluido en el marco que define el individuo correspondiente a esta mejor solución.

La siguiente imagen donde nos detenemos a hacer una revisión, es la de índice 79 dentro del conjunto de imágenes; Ilustración 5.4.6-2: DIARETDB1 resultado en imagen de retina, índice 80. Lo relevante de esta imagen es que la solución encontrada está a una distancia similar a al promedio de ICD encontrado para este experimento.



**Ilustración 5.4.6-2: DIARETDB1 resultado en imagen de retina, índice 80**

Por poner un caso donde la solución no fue encontrada, es seleccionada la imagen de índice 84 del *dataset*, Ilustración 5.4.6-3: DIARETDB1 resultado en imagen de retina, índice 84.

Es probable que sea una en la que los resultados en términos de *fitness*, MBF y SR hayan sido los menos óptimos.

El hecho de que en la imagen se visualice el “centro” del individuo con un desplazamiento respecto al marco, en magenta, es porque no ha sido tenido en cuenta el factor de escalado a la hora de dibujar el marco sobre la imagen de retina.

No obstante, para ser la mayor distancia intercentral encontrada (140), no es posible determinar una razón a priori observando la ilustración que pueda explicar este error.

Podría ser que la única razón de un resultado así esté únicamente relacionada con la naturaleza estocástica de *Differential Evolution*.



**Ilustración 5.4.6-3: DIARETDB1 resultado en imagen de retina, índice 84**

En general los resultados en términos de Tasa de Éxito para esta primera aplicación, tratándose de la implementación más sencilla del método diseñado, son lo suficientemente razonables como para considerar que merecería la pena profundizar en investigar sobre la utilización del método en otros conjuntos de imágenes.

Por supuesto, los resultados obtenidos no son competitivos con otras aproximaciones presentes en la literatura aplicadas concretamente a DIARETDB1.



## **CAPÍTULO 6. CONCLUSIONES Y TRABAJO FUTURO**

---

## 6.1. Conclusiones

Este estudio ha consistido en el diseño y validación de un método basado en el Operador LBP y *Differential Evolution* para el problema de reconocimiento o detección de objetos, característico de la Visión Artificial.

Para problemas de localización de un template en una escena, donde el mismo template está presente en la escena sin haber sufrido transformaciones afines, ruido ni conversiones de ningún tipo, el método ha exhibido en los experimentos presentados una precisión y estabilidad muy elevada, constatando que el método es muy superior a realizar una búsqueda por fuerza bruta en términos de velocidad.

Para problemas donde puede aparecer un factor de escalado para la correspondencia absoluta entre template y objeto en la escena, el método consume algo más de tiempo y pierde ligeramente algo de precisión, sin embargo continúa obteniendo unas tasas de éxito razonablemente buenas de forma habitual.

En los casos donde el template y el objeto se diferencian por existir una rotación en el template, el tiempo que el método tarda en encontrar soluciones aumenta y aumentan las posibilidades de caer en óptimos locales.

Los problemas estándar donde intervienen tanto cambio de escala como rotación, además de volver a aumentar el tiempo de ejecución, la precisión comienza a notarse especialmente sensible al tipo de template; según su homogeneidad, que hay que localizar en una escena.

Esto es un indicativo de que el método está condicionado por la relación existente entre el tipo de textura del template y el tipo de textura general de la escena.

Además, se ha podido comprobar que la implementación esencial es lo suficientemente potente como no necesitarse recurrir a la aplicación de mejoras tradicionales para los algoritmos evolutivos y otras Metaheurísticas, ya que la introducción de dichas mejoras, como búsqueda local o variaciones de la función de evaluación ha producido unos resultados discretamente mejores en términos de precisión a costa de una pérdida de eficiencia computacional que será imprescindible en numerosas hipotéticas aplicaciones.

Se consolida una vez más la capacidad del Operador LBP para tolerar cambios de intensidad luminosa. Esto supone un gran beneficio y abre muchas vías de mejora ya que gracias a esta característica es posible prescindir de mecanismos adicionales para solventar problemas donde existe una variación uniforme de la intensidad luminosa, como por ejemplo introducir dimensiones adicionales en la representación del individuo.

Se ha demostrado que la metodología es aplicable al dominio del problema de la detección de objetos en imágenes digitales, y aplicable a casos reales.

## 6.2. Trabajo Futuro

Las versiones de DE y LBP empleadas en la metodología analizada en este estudio han sido acotadas a sus formatos más básicos. No se ha indagado por motivos de alcance en analizar otras posibles estrategias disponibles en la DE, ni se han utilizado otras variantes del Operador LBP, como el invariante a rotación ni otras versiones más modernas de los últimos años.

Tan solo para estos dos elementos, el posible trabajo futuro es francamente amplio. Podrían probarse numerosas combinaciones de DE y LBP, con distintas combinaciones de sus parámetros internos de cada algoritmo.

El reajuste del histograma LBP es un camino que podría mejorar la eficacia del método para detectar objetos basándose en su textura. Concretamente, ya que la última posición del vector del descriptor almacena todos los valores “uniformes” en la versión LBPu del algoritmo, quizá la supresión de esa posición podría contribuir a enfatizar las texturas particulares de cada textura y por lo tanto aumentar la precisión del método.

Otras posibilidades bien pudieran orientarse más a aplicar herramientas de la Visión Artificial, desde aplicar etapas de preprocesamiento de las imágenes como la aplicación de filtros o utilización de otros descriptores junto al motor de búsqueda DE, o bien podría investigarse la aplicación de otras Metaheurísticas junto al operador LBP sustituyendo a DE como motor de búsqueda.

La aplicación de conocimiento del dominio propio del problema en cuestión que pretenda ser resuelto en base a esta metodología es una opción a la que no se ha recurrido en este estudio pero que podría resolver algunas de las cuestiones y dificultades halladas sobre la marcha.

Dado que se aprecia la existencia de una relación clara entre la textura del template y la textura de la escena, incorporar algún mecanismo para codificar e incorporar información global de la escena que sirva para implementar una estrategia de inicialización de la población inicial, podría contribuir tanto a la precisión como a la velocidad de convergencia.

En definitiva, este estudio podría seguir desarrollándose siguiendo diferentes caminos, y mientras el problema de la detección de objetos siga siendo un campo de estudio, este tipo de propuestas seguirán teniendo interés de investigación.





## REFERENCIAS

---

- Alahi, A., Ortiz, R., & Vandergheynst, P. (2012). Freak: Fast retina keypoint. *Ieee*, (págs. 510-517).
- Back, T., Hoffmeister, F., & Schwefel, H.-P. (1991). A survey of evolution strategies. *Morgan Kaufmann Publishers, San Mateo*, 2.
- Bay, H., Ess, A., Tuytelaars, T., & Gool, L. V. (2008). Speeded-up robust features (SURF). *Computer vision and image understanding*, 110(3), 346-359.
- Billinghurst, M., Clark, A., Lee, G., & others. (2015). A survey of augmented reality. *Foundations and Trends{\textregistered} in Human--Computer Interaction*, 8(2-3), 73-272.
- Brahnam, S., Jain, L. C., Nanni, L., Lumini, A., & others. (2014). *Local binary patterns: new variants and applications*. Springer.
- Busso, C., Deng, Z., Yildirim, S., Bulut, M., Lee, C. M., Kazemzadeh, A., . . . Narayanan, S. (2004). Analysis of emotion recognition using facial expressions, speech and multimodal information. *ACM*, (págs. 205-211).
- Cagnoni, S., Lutton, E., & Olague, G. (2007). *Genetic and evolutionary computation for image processing and analysis*. Hindawi Publishing Corporation.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*(6), 679-698.
- Carmona, E. J., Rincón, M., García-Feijóo, J., & Martínez-de-la-Casa, J. M. (2008). Identification of the optic nerve head with genetic algorithms. *Artificial Intelligence in Medicine*, 43(3), 243-259.
- Centeno, T. M., Lopes, H. S., Felisberto, M. K., & Ramos, L. V. (2005). Object detection for computer vision using a robust genetic algorithm. *Springer*, (págs. 284-293).
- Chen, C., Jafari, R., & Kehtarnavaz, N. (2015). Action recognition from depth sequences using depth motion maps-based local binary patterns. *IEEE*, (págs. 1092-1099).
- Chidambaram, C., & Lopes, H. S. (2009). A new approach for template matching in digital images using an artificial bee colony algorithm. *IEEE*, (págs. 146-151).
- Dalai, N. (2005). Histograms of oriented gradients for human detection., (págs. 886-893).
- Das, S., Mullick, S. S., & Suganthan, P. N. (2016). Recent advances in differential evolution--an updated survey. *Swarm and Evolutionary Computation*, 27, 1-30.

- Duda, R. O., & Hart, P. E. (1972). Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1), 11-15.
- Eiben, A. E., & Smith, J. E. (2003). *Introduction to evolutionary computing* (Vol. 53). Springer.
- Galbally, J., Ross, A., Gomez-Barrero, M., Fierrez, J., & Ortega-Garcia, J. (2013). Iris image reconstruction from binary templates: An efficient probabilistic approach based on genetic algorithms. *Computer Vision and Image Understanding*, 117(10), 1512-1525.
- Haleem, M. S., Han, L., Hemert, J. v., & Li, B. (2013). Automatic extraction of retinal features from colour retinal images for glaucoma diagnosis: a review. *Computerized medical imaging and graphics*, 37(7), 581-596.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems. An introductory analysis with application to biology, control, and artificial intelligence*. Ann Arbor, MI: University of Michigan Press.
- Kennedy, J. (2006). Swarm intelligence. En *Handbook of nature-inspired and innovative computing* (págs. 187-219). Springer.
- Khan, M. W. (2014). A survey: image segmentation techniques. *International Journal of Future Computer and Communication*, 3(2), 89.
- Koza, J. R. (1992). *Genetic programming: on the programming of computers by means of natural selection* (Vol. 1). MIT press.
- Learned-Miller, E. G. (2011). *Introduction to computer vision*. University of Massachusetts, Amherst.
- Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011). BRISK: Binary robust invariant scalable keypoints. *IEEE*, (págs. 2548-2555).
- Li, W., Chen, C., Su, H., & Du, Q. (2015). Local binary patterns and extreme learning machine for hyperspectral imagery classification. *IEEE Transactions on Geoscience and Remote Sensing*, 53(7), 3681-3693.
- Lutton, E., & Martinez, P. (1994). A genetic algorithm for the detection of 2D geometric primitives in images. *IEEE*, 1, págs. 526-528.
- Malamas, E. N., Petrakis, E. G., Zervakis, M., Petit, L., & Legat, J.-D. (2003). A survey on industrial vision systems, applications and tools. *Image and vision computing*, 21(2), 171-188.

- Matas, J., Chum, O., Urban, M., & Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and vision computing*, 22(10), 761-767.
- Mathworks. (01 de 06 de 2017). *Mathworks Matlab % Simulink*. Obtenido de Random Numbers from Normal Distribution with Specific Mean and Variance: <https://es.mathworks.com/help/matlab/math/random-numbers-with-specific-mean-and-variance.html>
- Mira, J., Martínez, R., Rincón, M., Bachiller, M., & Fernández-Caballero, A. (2007). Towards a Semi-automatic Situation Diagnosis System in Surveillance Tasks. *Springer*, (págs. 90-98).
- Muggleton-Harris, A., & Brown, J. (1988). Cytoplasmic factors influence mitochondrial reorganization and resumption of cleavage during culture of early mouse embryos. *Human Reproduction*, 3(8), 1020-1028.
- Murala, S., & Wu, Q. J. (2014). Local mesh patterns versus local binary patterns: biomedical image indexing and retrieval. *IEEE Journal of Biomedical and Health Informatics*, 18(3), 929-938.
- Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7), 971-987.
- Patil, J. K., & Kumar, R. (2011). Advances in image processing for detection of plant diseases. *Journal of Advanced Bioinformatics Applications and Research*, 2(2), 135-141.
- Paulinas, M., & Ušinskas, A. (2015). A survey of genetic algorithms applications for image enhancement and segmentation. *Information Technology and control*, 36(3).
- Perlin, H. A., Lopes, H. S., & Centeno, T. M. (2008). Particle swarm optimization for object recognition in computer vision. *Springer*, (págs. 11-21).
- Pietikäinen, M., & Zhao, G. (2015). Two decades of local binary patterns: A survey. *Advances in Independent Component Analysis and Learning Machines*, 175-210.
- Qi, X., Xiao, R., Li, C.-G., Qiao, Y., Guo, J., & Tang, X. (2014). Pairwise rotation invariant co-occurrence local binary pattern. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11), 2199-2213.
- Rechenberg, I. (1972). Evolution Strategy: Optimizations of Technical Systems Based on Principles of Biological Evolution. *Frommann-Holzboog, Stuttgart*.

- Roberts, L. G. (1963). *Machine perception of three-dimensional solids*. Ph.D. dissertation, Massachusetts Institute of Technology.
- Rosten, E., & Drummond, T. (2005). Fusing points and lines for high performance tracking. *IEEE*, 2, págs. 1508-1515.
- Ryan, C., Collins, J., & Neill, M. O. (1998). Grammatical evolution: Evolving programs for an arbitrary language. *Springer*, (págs. 83-96).
- Shi, J., & others. (1994). Good features to track. *IEEE*, (págs. 593-600).
- Silva, R. R., Lopes, H. S., & Erig, C. R. (2008). A compact genetic algorithm with elitism and mutation applied to image recognition. *Springer*, (págs. 1109-1116).
- Smara, G. A., & Khalefah, F. (2014). Localization of license plate number using dynamic image processing techniques and genetic algorithms. *IEEE transactions on evolutionary computation*, 18(2), 244-257.
- Sobel, M. E. (1982). Asymptotic confidence intervals for indirect effects in structural equation models. *Sociological methodology*, 13, 290-312.
- Sonka, M., Hlavac, V., & Boyle, R. (2014). *Image processing, analysis, and machine vision*. Cengage Learning.
- Storn, R., & Price, K. (1997). Differential evolution--a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4), 341-359.
- Swets, D. L., Punch, B., & Weng, J. (1995). Genetic algorithms for object recognition in a complex scene. *IEEE*, 2, págs. 595-598.
- Trujillo, L., & Olague, G. (2006). Synthesis of interest point detectors through genetic programming. *ACM*, (págs. 887-894).
- Ugolotti, R., Nashed, Y. S., Mesejo, P., Ivekovi{v{c}}, {., Mussi, L., & Cagnoni, S. (2013). Particle swarm optimization and differential evolution for model-based object detection. *Applied Soft Computing*, 13(6), 3092-3105.
- Viergever, M. A., Maintz, J. A., Klein, S., Murphy, K., Staring, M., & Pluim, J. P. (2016). A survey of medical image registration--under review. *A survey of medical image registration--under review*. Elsevier.
- Zhang, J. W., & Wang, G. G. (2012). Image matching using a bat algorithm with mutation. *Trans Tech Publ*, 203, págs. 88-93.











# Anexo A: Tablas de Resultados

## A.1: Resultados Decodificación Básica

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	2	20	50	50	1	6.3127	0.0034335 ± 0.00019444	20	47.9131	4.7913 ± 0.075892	0.003372
0.2	0.5	2	20	50	50	1	5.4231	0.003372 ± 0	26	48.1848	4.8185 ± 0.056721	0.003372
0.2	0.8	2	20	50	50	1	4.7874	0.0080675 ± 0.0081142	176	47.6394	4.7639 ± 0.04584	0.0034215
0.5	0.2	2	20	50	50	1	6.0415	0.0033788 ± 2.1379e-05	20	47.5285	4.7528 ± 0.042553	0.003372
0.5	0.5	2	20	50	50	1	5.3688	0.0034097 ± 0.00010306	90	47.6462	4.7646 ± 0.063641	0.003372
0.5	0.8	2	20	50	50	1	5.3035	0.0056878 ± 0.0018562	288	47.6225	4.7622 ± 0.06093	0.0034215
0.8	0.2	2	20	50	50	1	5.8688	0.003372 ± 0	26	47.2507	4.7251 ± 0.015604	0.003372
0.8	0.5	2	20	50	50	1	6.2819	0.0035817 ± 0.00020047	168	47.3279	4.7328 ± 0.038088	0.003372
0.8	0.8	2	20	50	50	1	1.5726	0.0072271 ± 0.0034546	188	48.0986	4.8099 ± 0.05888	0.0039166

Tabla A.1-1: Resultados decodificación básica TLF

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	3	30	150	50	1	1.1765	0.0069969 ± 0.0021849	30	150.2892	15.0289 ± 0.22453	0.0048885
0.2	0.5	3	30	150	50	1	1.2887	0.0094678 ± 0.0051875	447	150.0854	15.0085 ± 0.068246	0.0048885
0.2	0.8	3	30	150	50	1	6.4637	0.008991 ± 0.00073081	555	150.2744	15.0274 ± 0.02574	0.0081079
0.5	0.2	3	30	150	50	1	1.4562	0.0072218 ± 0.00093006	393	149.7023	14.9702 ± 0.12081	0.0049325
0.5	0.5	3	30	150	50	1	1.59	0.0075431 ± 0.00091585	537	149.7568	14.9757 ± 0.020123	0.0053004
0.5	0.8	3	30	150	50	1	5.3418	0.0094722 ± 0.0015227	420	150.1489	15.0149 ± 0.029815	0.0081136
0.8	0.2	3	30	150	50	1	5.6023	0.0078628 ± 0.00025719	516	149.7664	14.9766 ± 0.023424	0.0075543
0.8	0.5	3	30	150	50	1	1.508	0.0077269 ± 0.00046883	633	150.0559	15.0056 ± 0.067512	0.0066297
0.8	0.8	3	30	150	50	1	6.6308	0.01005 ± 0.00062941	498	150.2651	15.0265 ± 0.037334	0.0088345

Tabla A.1-2: Resultados decodificación básica TLFs

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	3	30	150	50	1	1.8407	0.0047811 ± 0.0012324	93	265.6475	26.5647 ± 0.23277	0.0040551
0.2	0.5	3	30	150	50	1	2.0348	0.0056667 ± 0.0016254	429	266.1955	26.6196 ± 0.26333	0.0040551
0.2	0.8	3	30	150	50	1	1.6108	0.0072594 ± 0.0019993	471	265.176	26.5176 ± 0.30822	0.0052318
0.5	0.2	3	30	150	50	1	4.5924	0.004972 ± 0.0012949	513	266.2797	26.628 ± 0.23999	0.0041732
0.5	0.5	3	30	150	50	1	4.7523	0.0054604 ± 0.00014272	372	263.9101	26.391 ± 0.33932	0.0041732
0.5	0.8	3	30	150	50	1	5.7359	0.007715 ± 0.00085392	477	266.4215	26.6422 ± 0.22499	0.0068042
0.8	0.2	3	30	150	50	1	3.7791	0.0058093 ± 0.0011359	381	263.143	26.3143 ± 0.18311	0.0049747
0.8	0.5	3	30	150	50	1	5.086	0.005832 ± 0.0011604	513	262.2121	26.2212 ± 0.39249	0.0042821
0.8	0.8	3	30	150	50	1	4.3077	0.0080482 ± 0.00047629	534	263.8533	26.3853 ± 0.30966	0.0073177

Tabla A.1-3: Resultados decodificación básica TLFr

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	1	5.5334	0.0081369 ± 0.0013418	316	449.3586	44.9359 ± 0.2494	0.0066692
0.2	0.5	4	40	250	50	1	3.4187	0.0077437 ± 0.00026668	844	451.0533	45.1053 ± 0.17174	0.0073828
0.2	0.8	4	40	250	50	1	1.1183	0.0089794 ± 0.00054772	792	454.0293	45.4029 ± 0.24713	0.0083497
0.5	0.2	4	40	250	50	1	1.4416	0.0086275 ± 0.0011283	1000	451.9128	45.1913 ± 0.20519	0.0079231
0.5	0.5	4	40	250	50	1	1.3823	0.0080326 ± 0.00023268	400	452.2677	45.2268 ± 0.28809	0.0077079
0.5	0.8	4	40	250	50	1	7.1175	0.010342 ± 0.0015301	1100	452.3103	45.231 ± 0.2754	0.0089283
0.8	0.2	4	40	250	50	1	5.8232	0.008489 ± 0.0010555	756	447.1563	44.7156 ± 0.28019	0.0076254
0.8	0.5	4	40	250	50	1	7.0495	0.0083572 ± 0.00032429	880	448.0592	44.8059 ± 0.2328	0.0077935
0.8	0.8	4	40	250	50	1	8.9422	0.010814 ± 0.00061887	720	449.2373	44.9237 ± 0.18679	0.010002

Tabla A.1-4: Resultados decodificación básica TLFsr

## A.2: Resultados Exploración templates homogéneos

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.4	0.89053	0.020685 ± 0.0093384	520	378.7252	37.8725 ± 0.52237	0.0081546
0.2	0.5	4	40	250	50	0.6	0.47311	0.018739 ± 0.011643	1033.3333	378.0856	37.8086 ± 0.30169	0.009135
0.2	0.8	4	40	250	50	0.7	2.7655	0.025109 ± 0.0083529	697.1429	375.605	37.5605 ± 0.24698	0.012679
0.5	0.2	4	40	250	50	0.6	4.9099	0.017298 ± 0.0099245	840	375.3425	37.5342 ± 0.24174	0.0093102
0.5	0.5	4	40	250	50	0.9	1.1877	0.015626 ± 0.0085431	662.2222	370.3199	37.032 ± 0.28477	0.0091883
0.5	0.8	4	40	250	50	0.5	22.4646	0.035197 ± 0.0069217	776	369.8668	36.9867 ± 0.45523	0.02302
0.8	0.2	4	40	250	50	1	2.4599	0.010714 ± 0.00047884	664	367.8771	36.7877 ± 0.43327	0.0098512
0.8	0.5	4	40	250	50	1	6.0865	0.022025 ± 0.0080976	852	361.5858	36.1586 ± 0.44314	0.011957
0.8	0.8	4	40	250	50	0.5	3.8573	0.0342 ± 0.0081074	560	365.5735	36.5573 ± 0.28056	0.016645

Tabla A.2-1: Resultados exploración templates homogéneos, pilas GR=250

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.3	3.2654	0.028787 ± 0.01179	533.3333	317.9658	31.7966 ± 0.4667	0.011693
0.2	0.5	4	40	250	50	0.6	1.941	0.02565 ± 0.012234	953.3333	326.8226	32.6823 ± 0.76044	0.012736
0.2	0.8	4	40	250	50	0.5	2.7498	0.037507 ± 0.0079889	904	318.4765	31.8476 ± 0.52209	0.022787
0.5	0.2	4	40	250	50	0.5	2.2356	0.026233 ± 0.011857	960	321.3447	32.1345 ± 0.92733	0.012169
0.5	0.5	4	40	250	50	0.7	7.2105	0.032136 ± 0.010196	862.8571	313.5276	31.3528 ± 0.66891	0.017029
0.5	0.8	4	40	250	50	0.5	7.3717	0.038718 ± 0.0087764	960	313.5067	31.3507 ± 0.61804	0.022833
0.8	0.2	4	40	250	50	0.7	1.8071	0.03081 ± 0.0097302	1051.4286	301.8653	30.1865 ± 0.77472	0.01565
0.8	0.5	4	40	250	50	0.6	50.1066	0.031156 ± 0.0082537	746.6667	305.5651	30.5565 ± 0.66886	0.023069
0.8	0.8	4	40	250	50	0.4	12.0206	0.038196 ± 0.01004	1210	310.8847	31.0885 ± 0.42802	0.026174

Tabla A.2-2: Resultados exploración templates homogéneos, rueda GR=250

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	350	50	0.4	1.1988	0.022243 ± 0.011702	40	554.3524	55.4352 ± 0.59435	0.0081546
0.2	0.5	4	40	350	50	0.6	0.98493	0.017468 ± 0.010487	940	552.6979	55.2698 ± 0.49494	0.0089462
0.2	0.8	4	40	350	50	0.5	8.235	0.026716 ± 0.010807	816	549.0386	54.9039 ± 0.55898	0.014944

0.5	0.2	4	40	350	50	0.7	3.2274	0.014644 ± 0.0082166	954.2857	551.0136	55.1014 ± 0.39045	0.0089564
0.5	0.5	4	40	350	50	0.8	0.51834	0.015106 ± 0.0093261	740	543.1605	54.3161 ± 0.56536	0.010219
0.5	0.8	4	40	350	50	0.9	7.871	0.022479 ± 0.0083445	897.7778	540.4218	54.0422 ± 0.48303	0.014633
0.8	0.2	4	40	350	50	1	3.2344	0.010023 ± 0.00041889	852	540.9169	54.0917 ± 0.59345	0.009356
0.8	0.5	4	40	350	50	1	6.4772	0.017394 ± 0.0069392	1072	513.1058	51.3106 ± 0.86694	0.010751
0.8	0.8	4	40	350	50	0.9	12.9059	0.02558 ± 0.0063571	800	525.5951	52.5595 ± 0.68136	0.018417

Tabla A.2-3: Resultados exploración templates homogéneos, pilas GR=350

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	350	50	0.3	1.5452	0.028817 ± 0.011544	40	470.385	47.0385 ± 0.47323	0.011568
0.2	0.5	4	40	350	50	0.3	2.121	0.028979 ± 0.01135	466.6667	463.6018	46.3602 ± 1.2486	0.012204
0.2	0.8	4	40	350	50	0.6	7.2891	0.033117 ± 0.0083387	946.6667	457.4743	45.7474 ± 0.37287	0.021843
0.5	0.2	4	40	350	50	0.3	2.5258	0.029706 ± 0.01034	693.3333	465.9674	46.5967 ± 0.55533	0.012324
0.5	0.5	4	40	350	50	0.6	5.7809	0.029225 ± 0.0096423	733.3333	457.3848	45.7385 ± 0.97802	0.014533
0.5	0.8	4	40	350	50	0.6	4.5029	0.033677 ± 0.0096563	600	454.3722	45.4372 ± 0.57027	0.022697
0.8	0.2	4	40	350	50	0.9	3.0124	0.014998 ± 0.0026096	1124.4444	444.4892	44.4489 ± 0.79857	0.013284
0.8	0.5	4	40	350	50	0.9	5.7673	0.028268 ± 0.0083809	706.6667	428.362	42.8362 ± 1.0459	0.018973
0.8	0.8	4	40	350	50	0.5	1.3579	0.036139 ± 0.0096265	848	442.7056	44.2706 ± 1.1675	0.020699

Tabla A.2-4: Resultados exploración templates homogéneos, rueda GR=350

### A.3: Resultados Influencia de la intensidad luminosa

Resultados escena  $\gamma = 2.0$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0	53.7921	0.02021 ± 0.0013918	0	500.5548	50.0555 ± 0.94985	0.018435
0.2	0.5	4	40	250	50	0	61.5466	0.02206 ± 0.0025208	0	490.213	49.0213 ± 0.34756	0.019514
0.2	0.8	4	40	250	50	0	59.5762	0.022881 ± 0.00049458	0	492.16	49.216 ± 0.22413	0.022268
0.5	0.2	4	40	250	50	0	58.4195	0.021579 ± 0.00090665	0	492.7204	49.272 ± 0.3257	0.02105
0.5	0.5	4	40	250	50	0	59.5186	0.021227 ± 0.00059739	0	489.9342	48.9934 ± 0.16338	0.020547
0.5	0.8	4	40	250	50	0	68.9943	0.024217 ± 0.00067257	0	492.9499	49.295 ± 0.34006	0.022842
0.8	0.2	4	40	250	50	0	62.3703	0.021849 ± 0.00030481	0	486.4961	48.6496 ± 0.11353	0.021265
0.8	0.5	4	40	250	50	0	65.8948	0.022173 ± 0.00048699	0	486.6604	48.666 ± 0.31142	0.021458
0.8	0.8	4	40	250	50	0	66.1533	0.024947 ± 0.00081671	0	489.6072	48.9607 ± 0.35038	0.023734

Tabla A.3-1: Resultados influencia de la intensidad luminosa, circuito  $\gamma=2.0$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.2	41.8878	0.036366 ± 0.011985	40	413.5771	41.3577 ± 0.3156	0.022396
0.2	0.5	4	40	250	50	0.1	37.8819	0.034197 ± 0.0052976	1440	414.0123	41.4012 ± 0.21918	0.024854
0.2	0.8	4	40	250	50	0.3	25.1815	0.042482 ± 0.0048262	893.3333	411.1958	41.1958 ± 0.38606	0.036093
0.5	0.2	4	40	250	50	0	812.5944	0.033849 ± 0.0080835	0	412.2349	41.2235 ± 0.28129	0.030304
0.5	0.5	4	40	250	50	0.3	16.3518	0.037703 ± 0.0056409	666.6667	408.8559	40.8856 ± 0.409	0.0315
0.5	0.8	4	40	250	50	0.2	18.8895	0.04332 ± 0.0047443	620	407.7821	40.7782 ± 0.42789	0.034003
0.8	0.2	4	40	250	50	0.1	54.5557	0.030837 ± 0.0022862	1240	402.1078	40.2108 ± 0.78127	0.026508
0.8	0.5	4	40	250	50	0.3	41.4615	0.041668 ± 0.0036082	800	395.4528	39.5453 ± 0.69959	0.036242
0.8	0.8	4	40	250	50	0.1	823.6211	0.043348 ± 0.0061081	600	401.4218	40.1422 ± 0.52198	0.036429

Tabla A.3-2: Resultados influencia de la intensidad luminosa, pilas  $\gamma=2.0$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0	300.1335	0.037826 ± 0.0027005	0	341.9913	34.1991 ± 0.48879	0.035044
0.2	0.5	4	40	250	50	0.3	1.3789	0.034227 ± 0.012115	626.6667	340.8185	34.0819 ± 0.40292	0.015479
0.2	0.8	4	40	250	50	0.3	6.5119	0.043502 ± 0.0084461	506.6667	342.7467	34.2747 ± 0.53335	0.026351
0.5	0.2	4	40	250	50	0.1	34.9695	0.038786 ± 0.0064256	1200	341.0822	34.1082 ± 0.63743	0.021394
0.5	0.5	4	40	250	50	0.2	7.6764	0.042065 ± 0.0087029	1220	336.5819	33.6582 ± 0.63108	0.02569
0.5	0.8	4	40	250	50	0.5	8.2263	0.042534 ± 0.0052137	688	333.5911	33.3591 ± 0.44414	0.032991
0.8	0.2	4	40	250	50	0.4	1.222	0.037834 ± 0.012308	870	324.0841	32.4084 ± 0.77995	0.015501
0.8	0.5	4	40	250	50	0.6	2.1257	0.039878 ± 0.0086674	960	325.8341	32.5834 ± 0.46396	0.022802
0.8	0.8	4	40	250	50	0.1	593.9377	0.048921 ± 0.0042514	1240	326.1578	32.6158 ± 0.42531	0.042781

Tabla A.3-3: Resultados influencia de la intensidad luminosa, rueda  $\gamma=2.0$

Resultados escena  $\gamma = 1.0$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	1	4.4725	0.007958 ± 0.002063	380	500.4364	50.0436 ± 2.3859	0.0059555
0.2	0.5	4	40	250	50	1	4.332	0.0079632 ± 0.0011497	740	497.2218	49.7222 ± 0.5768	0.0067952
0.2	0.8	4	40	250	50	0.9	1.1485	0.011573 ± 0.0058325	964.44	495.8041	49.5804 ± 0.2419	0.0077975
0.5	0.2	4	40	250	50	1	3.6754	0.0077984 ± 0.00039059	884	491.1039	49.1104 ± 0.47259	0.0069412
0.5	0.5	4	40	250	50	1	2.3918	0.0080118 ± 0.00035136	584	491.3848	49.1385 ± 0.51578	0.0073682
0.5	0.8	4	40	250	50	1	7.4391	0.01013 ± 0.0017113	972	494.2566	49.4257 ± 0.28707	0.0079832
0.8	0.2	4	40	250	50	1	3.2303	0.0082721 ± 0.0002866	820	487.3701	48.737 ± 0.39532	0.0079581
0.8	0.5	4	40	250	50	1	2.5646	0.0083299 ± 0.00072447	804	487.3397	48.734 ± 0.48135	0.0074556
0.8	0.8	4	40	250	50	1	5.0736	0.011158 ± 0.0014246	996	493.516	49.3516 ± 0.50382	0.0096059

Tabla A.3-4: Resultados influencia de la intensidad luminosa, circuito  $\gamma=1.0$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± MBF STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.2	3.0017	0.026294 ± 0.007303	40	421.5914	42.1591 ± 0.48105	0.013026
0.2	0.5	4	40	250	50	0.7	2.3982	0.019707 ± 0.0063173	468.5714	422.6787	42.2679 ± 0.3159	0.01465
0.2	0.8	4	40	250	50	0.8	7.9974	0.030365 ± 0.007613	695	418.3722	41.8372 ± 0.48002	0.02198
0.5	0.2	4	40	250	50	0.8	1.2117	0.018145 ± 0.0064639	920	417.7779	41.7778 ± 0.5242	0.014438
0.5	0.5	4	40	250	50	0.7	5.608	0.022504 ± 0.0084158	674.2857	417.5479	41.7548 ± 0.61142	0.014662
0.5	0.8	4	40	250	50	0.7	4.3796	0.033182 ± 0.0069318	925.7143	419.5119	41.9512 ± 0.87095	0.023762
0.8	0.2	4	40	250	50	1	4.022	0.016021 ± 0.00048776	692	412.2341	41.2234 ± 0.72459	0.015303
0.8	0.5	4	40	250	50	0.8	4.1157	0.027549 ± 0.0079693	860	404.0079	40.4008 ± 0.62929	0.018651
0.8	0.8	4	40	250	50	0.6	6.6707	0.032209 ± 0.0090174	826.6667	407.7073	40.7707 ± 0.88346	0.021353

**Tabla A.3-5: Resultados influencia de la intensidad luminosa, pilas  $\gamma=1.0$**

F	CR	D	NP	GR	SRT	SR	ICD	MBF $\pm$ MBF STD	AES	TOTAL TIME	TPR AVG $\pm$ STD	Best fitness
0.2	0.2	4	40	250	50	0.5	1.2255	0.024787 $\pm$ 0.012725	40	347.1815	34.7182 $\pm$ 2.9274	0.011847
0.2	0.5	4	40	250	50	0.5	5.3734	0.027728 $\pm$ 0.013006	880	346.9301	34.693 $\pm$ 0.33154	0.014639
0.2	0.8	4	40	250	50	0.5	2.4884	0.035164 $\pm$ 0.010282	848	344.1122	34.4112 $\pm$ 0.47493	0.021829
0.5	0.2	4	40	250	50	0.8	2.1352	0.02 $\pm$ 0.0093335	695	345.2596	34.526 $\pm$ 1.0698	0.014088
0.5	0.5	4	40	250	50	0.6	47.6741	0.035401 $\pm$ 0.0068377	1133.3333	335.2267	33.5227 $\pm$ 1.0715	0.024168
0.5	0.8	4	40	250	50	0.6	4.4252	0.039019 $\pm$ 0.0079793	940	340.2248	34.0225 $\pm$ 0.78292	0.025895
0.8	0.2	4	40	250	50	0.8	0.83346	0.022981 $\pm$ 0.010579	685	325.8166	32.5817 $\pm$ 1.2533	0.014569
0.8	0.5	4	40	250	50	0.9	3.2563	0.033629 $\pm$ 0.005841	862.2222	320.996	32.0996 $\pm$ 1.0091	0.025288
0.8	0.8	4	40	250	50	0.4	4.767	0.040441 $\pm$ 0.0079686	870	329.8184	32.9818 $\pm$ 0.7101	0.027777

**Tabla A.3-6: Resultados influencia de la intensidad luminosa, rueda  $\gamma=1.0$**

### Resultados escena $\gamma = 0.5$

F	CR	D	NP	GR	SRT	SR	ICD	MBF $\pm$ MBF STD	AES	TOTAL TIME	TPR AVG $\pm$ STD	Best fitness
0.2	0.2	4	40	250	50	1	6.029	0.011534 $\pm$ 0.0003064	596	495.188	49.5188 $\pm$ 0.24955	0.01116
0.2	0.5	4	40	250	50	1	0.97377	0.013106 $\pm$ 0.0010119	780	500.4424	50.0442 $\pm$ 0.4006	0.012084
0.2	0.8	4	40	250	50	1	4.9337	0.014162 $\pm$ 0.0010152	740	494.999	49.4999 $\pm$ 0.33285	0.013057
0.5	0.2	4	40	250	50	1	4.1059	0.012915 $\pm$ 0.0003347	684	493.4308	49.3431 $\pm$ 0.46571	0.012476
0.5	0.5	4	40	250	50	1	2.3041	0.013108 $\pm$ 0.0005611	1200	496.3724	49.6372 $\pm$ 0.70644	0.01202
0.5	0.8	4	40	250	50	1	3.0834	0.015492 $\pm$ 0.0015227	932	495.6728	49.5673 $\pm$ 0.27271	0.013889
0.8	0.2	4	40	250	50	1	8.0178	0.013566 $\pm$ 0.0002637	720	482.3524	48.2352 $\pm$ 0.8929	0.013103
0.8	0.5	4	40	250	50	1	6.1083	0.013481 $\pm$ 0.0003629	764	478.4364	47.8436 $\pm$ 0.43259	0.013067
0.8	0.8	4	40	250	50	1	5.8198	0.016757 $\pm$ 0.001491	832	482.6272	48.2627 $\pm$ 0.50189	0.014975

**Tabla A.3-7: Resultados influencia de la intensidad luminosa, circuito  $\gamma=0.5$**

F	CR	D	NP	GR	SRT	SR	ICD	MBF $\pm$ MBF STD	AES	TOTAL TIME	TPR AVG $\pm$ STD	Best fitness
0.2	0.2	4	40	250	50	0.4	8.1908	0.025704 $\pm$ 0.0096017	810	432.3714	43.2371 $\pm$ 3.4693	0.014398
0.2	0.5	4	40	250	50	0.5	8.9015	0.024906 $\pm$ 0.0090101	544	409.9041	40.9904 $\pm$ 0.51525	0.015507
0.2	0.8	4	40	250	50	0.4	3.5843	0.035654 $\pm$ 0.0084331	340	407.3449	40.7345 $\pm$ 0.69959	0.020818
0.5	0.2	4	40	250	50	0.5	8.7828	0.023421 $\pm$ 0.0082247	912	399.6846	39.9685 $\pm$ 0.9006	0.014875
0.5	0.5	4	40	250	50	0.8	12.0211	0.022684 $\pm$ 0.0086447	640	389.3342	38.9334 $\pm$ 0.42442	0.01626
0.5	0.8	4	40	250	50	0.7	6.4847	0.035183 $\pm$ 0.0096044	851.43	391.9803	39.198 $\pm$ 0.6014	0.022644
0.8	0.2	4	40	250	50	0.8	4.9617	0.02016 $\pm$ 0.0070123	670	395.9229	39.5923 $\pm$ 0.63703	0.016595
0.8	0.5	4	40	250	50	0.8	8.7616	0.026844 $\pm$ 0.0072543	895	390.8492	39.0849 $\pm$ 0.58621	0.018306
0.8	0.8	4	40	250	50	0.7	15.5998	0.033037 $\pm$ 0.0078251	960	389.4569	38.9457 $\pm$ 0.47303	0.023676

**Tabla A.3-8: Resultados influencia de la intensidad luminosa, pilas  $\gamma=0.5$**

F	CR	D	NP	GR	SRT	SR	ICD	MBF $\pm$ MBF STD	AES	TOTAL TIME	TPR AVG $\pm$ STD	Best fitness
0.2	0.2	4	40	250	50	0.2	3.7246	0.032452 $\pm$ 0.0087774	800	319.5926	31.9593 $\pm$ 1.9411	0.016105
0.2	0.5	4	40	250	50	0.5	5.8385	0.0312 $\pm$ 0.011003	728	327.483	32.7483 $\pm$ 0.69511	0.018796
0.2	0.8	4	40	250	50	0.3	10.7606	0.039992 $\pm$ 0.0057616	640	331.1036	33.1104 $\pm$ 0.4591	0.028646
0.5	0.2	4	40	250	50	0.6	4.3638	0.025782 $\pm$ 0.0077232	686.7	330.563	33.0563 $\pm$ 0.95141	0.018205
0.5	0.5	4	40	250	50	0.4	11.6694	0.039929 $\pm$ 0.0075805	760	323.9741	32.3974 $\pm$ 0.78908	0.02822
0.5	0.8	4	40	250	50	0.2	49.2311	0.041355 $\pm$ 0.0062398	1500	324.1584	32.4158 $\pm$ 0.47926	0.028005
0.8	0.2	4	40	250	50	0.8	3.3202	0.02611 $\pm$ 0.0077876	935	318.0445	31.8045 $\pm$ 0.76776	0.018407
0.8	0.5	4	40	250	50	0.6	3.6163	0.036897 $\pm$ 0.0085064	700	311.4888	31.1489 $\pm$ 0.81514	0.021669
0.8	0.8	4	40	250	50	0.4	9.3279	0.04197 $\pm$ 0.0069375	1030	315.7115	31.5712 $\pm$ 0.60096	0.028426

**Tabla A.3-9: Resultados influencia de la intensidad luminosa, rueda  $\gamma=0.5$**

## A.4: Resultados Influencia del ruido gaussiano

Aplicación de una variación de ruido gaussiano,  $\gamma = 10^{-3}$  a una escena original

F	CR	D	NP	GR	SRT	SR	ICD	MBF $\pm$ STD	AES	T. TIME	TPR AVG $\pm$ STD	Best fitness
0.2	0.2	4	40	250	50	1	6.867	0.0092058 $\pm$ 0.0015319	512	448.0435	44.8044 $\pm$ 0.17318	0.0078821
0.2	0.5	4	40	250	50	1	7.3669	0.0094002 $\pm$ 0.0010057	840	447.364	44.7364 $\pm$ 0.19515	0.0087679
0.2	0.8	4	40	250	50	1	6.9955	0.011721 $\pm$ 0.0017511	728	446.9554	44.6955 $\pm$ 0.24473	0.0096315
0.5	0.2	4	40	250	50	1	4.11	0.0095167 $\pm$ 0.0011502	620	446.2571	44.6257 $\pm$ 0.29213	0.008458
0.5	0.5	4	40	250	50	1	6.8712	0.0093575 $\pm$ 0.00038074	832	445.3886	44.5389 $\pm$ 0.22673	0.0087556
0.5	0.8	4	40	250	50	1	7.1147	0.013282 $\pm$ 0.0019193	616	443.1821	44.3182 $\pm$ 0.41154	0.011073
0.8	0.2	4	40	250	50	1	3.5542	0.010003 $\pm$ 0.0010657	972	437.8975	43.7897 $\pm$ 0.34865	0.0091046
0.8	0.5	4	40	250	50	1	7.6772	0.010068 $\pm$ 0.00027044	572	439.799	43.9799 $\pm$ 0.39073	0.0096379
0.8	0.8	4	40	250	50	1	13.6366	0.015459 $\pm$ 0.0014598	688	441.1916	44.1192 $\pm$ 0.48554	0.013364

**Tabla A.4-1: Resultados influencia ruido gaussiano, circuito  $\gamma=0.001$**

F	CR	D	NP	GR	SRT	SR	ICD	MBF $\pm$ STD	AES	T. TIME	TPR AVG $\pm$ STD	Best fitness
0.2	0.2	4	40	250	50	0.7	9.9465	0.022192 $\pm$ 0.014732	354.2857	386.1123	38.6112 $\pm$ 0.7962	0.01029
0.2	0.5	4	40	250	50	0.6	10.525	0.02206 $\pm$ 0.013748	1140	384.9632	38.4963 $\pm$ 0.59925	0.010504
0.2	0.8	4	40	250	50	0.7	19.1033	0.034227 $\pm$ 0.0091157	491.4286	375.2234	37.5223 $\pm$ 0.31204	0.019791
0.5	0.2	4	40	250	50	0.8	6.2082	0.018023 $\pm$ 0.011993	960	378.5365	37.8537 $\pm$ 0.57226	0.011034
0.5	0.5	4	40	250	50	0.6	8.5597	0.028656 $\pm$ 0.011603	993.3333	369.5824	36.9582 $\pm$ 0.44838	0.013359
0.5	0.8	4	40	250	50	0.6	15.6513	0.039007 $\pm$ 0.010257	713.3333	370.7018	37.0702 $\pm$ 0.51653	0.021688
0.8	0.2	4	40	250	50	0.8	7.3492	0.018013 $\pm$ 0.010874	855	375.6672	37.5667 $\pm$ 0.57285	0.012343
0.8	0.5	4	40	250	50	0.8	9.6742	0.024367 $\pm$ 0.0091119	475	365.6183	36.5618 $\pm$ 0.52369	0.012149
0.8	0.8	4	40	250	50	0.7	10.3833	0.032376 $\pm$ 0.0083627	965.7143	363.5186	36.3519 $\pm$ 0.61903	0.021688

**Tabla A.4-2: Resultados influencia ruido gaussiano, pilas  $\gamma=0.001$**

F	CR	D	NP	GR	SRT	SR	ICD	MBF $\pm$ STD	AES	T. TIME	TPR AVG $\pm$ STD	Best fitness
0.2	0.2	4	40	250	50	0.2	12.478	0.037293 $\pm$ 0.0080556	280	314.7058	31.4706 $\pm$ 0.65079	0.023162
0.2	0.5	4	40	250	50	0	497.6709	0.041859 $\pm$ 0.0036946	0	298.7118	29.8712 $\pm$ 1.0455	0.036572

0.2	0.8	4	40	250	50	0	501.6393	0.048614 ± 0.0024466	0	311.4971	31.1497 ± 0.44201	0.046173
0.5	0.2	4	40	250	50	0.1	70.198	0.037792 ± 0.0026967	840	283.3779	28.3378 ± 1.7112	0.034667
0.5	0.5	4	40	250	50	0	69.63	0.045528 ± 0.0023414	0	283.0998	28.31 ± 0.73003	0.041578
0.5	0.8	4	40	250	50	0.4	3.7551	0.047776 ± 0.0036356	1070	297.1112	29.7111 ± 0.50772	0.04101
0.8	0.2	4	40	250	50	0.1	9.6891	0.040981 ± 0.0049023	1400	286.9292	28.6929 ± 1.6759	0.029594
0.8	0.5	4	40	250	50	0	69.2621	0.044142 ± 0.0031275	0	268.5951	26.8595 ± 0.82558	0.038652
0.8	0.8	4	40	250	50	0.1	510.6977	0.048126 ± 0.0023198	280	290.9592	29.0959 ± 0.6163	0.043235

Tabla A.4-3: Resultados influencia ruido gaussiano, rueda  $\gamma=0.001$

Aplicación de una variación de ruido gaussiano,  $\gamma = 10^{-2}$  a una escena original

F	CR	D	N	GR	SR	SR	ICD	MBF ± STD	AES	T.TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	1	26.618	0.021714 ± 0.0007907	496	441.1647	44.1165 ± 0.71668	0.020688
0.2	0.5	4	40	250	50	1	23.723	0.022517 ± 0.00091594	528	439.4259	43.9426 ± 0.53346	0.021361
0.2	0.8	4	40	250	50	0.8	22.473	0.025331 ± 0.0019512	845	436.5889	43.6589 ± 0.45408	0.022093
0.5	0.2	4	40	250	50	0.7	33.297	0.023994 ± 0.0021715	662.8571	428.0008	42.8001 ± 0.43975	0.021449
0.5	0.5	4	40	250	50	1	29.262	0.025576 ± 0.0018835	864	428.2928	42.8293 ± 0.46757	0.021978
0.5	0.8	4	40	250	50	0.7	16.393	0.026789 ± 0.0017998	731.4286	432.9006	43.2901 ± 0.94813	0.023853
0.8	0.2	4	40	250	50	0.6	25.497	0.025449 ± 0.0031302	620	417.6366	41.7637 ± 0.42723	0.022208
0.8	0.5	4	40	250	50	0.7	28.095	0.02654 ± 0.0017542	845.7143	416.3678	41.6368 ± 0.20212	0.024156
0.8	0.8	4	40	250	50	0.7	54.479	0.026264 ± 0.0013142	800	422.8545	42.2854 ± 0.64454	0.024572

Tabla A.4-4: Resultados influencia ruido gaussiano, circuito  $\gamma=0.01$

F	CR	D	NP	GR	SR	SR	ICD	MBF ± STD	AES	T. TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.3	19.9939	0.054205 ± 0.011356	80	374.6133	37.4613 ± 0.64571	0.033357
0.2	0.5	4	40	250	50	0.3	20.6604	0.05149 ± 0.012107	746.6667	372.234	37.2234 ± 0.40314	0.035067
0.2	0.8	4	40	250	50	0.4	33.2237	0.060752 ± 0.0083345	890	371.8237	37.1824 ± 0.42643	0.04919
0.5	0.2	4	40	250	50	0.4	23.729	0.046659 ± 0.010343	680	374.0726	37.4073 ± 0.93153	0.036337
0.5	0.5	4	40	250	50	0.3	23.331	0.054806 ± 0.011945	466.6667	364.5753	36.4575 ± 0.93968	0.038702
0.5	0.8	4	40	250	50	0.8	9.1348	0.063177 ± 0.0058214	805	363.252	36.3252 ± 0.71535	0.05571
0.8	0.2	4	40	250	50	0.4	20.6558	0.057149 ± 0.01428	850	362.7789	36.2779 ± 2.135	0.038883
0.8	0.5	4	40	250	50	0.1	25.2326	0.062639 ± 0.009885	80	349.0559	34.9056 ± 1.3724	0.042006
0.8	0.8	4	40	250	50	0.3	54.125	0.065291 ± 0.0066928	800	358.1022	35.8102 ± 0.48755	0.055216

Tabla A.4-5: Resultados influencia ruido gaussiano, pilas  $\gamma=0.01$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	T. TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0	506.9354	0.048756 ± 0.0031763	0	302.9287	30.2929 ± 1.9561	0.045814
0.2	0.5	4	40	250	50	0	501.4808	0.050019 ± 0.0023068	0	284.5122	28.4512 ± 1.3076	0.047193
0.2	0.8	4	40	250	50	0.1	861.2739	0.055993 ± 0.001854	1360	302.6596	30.266 ± 0.72398	0.053703
0.5	0.2	4	40	250	50	0	507.3394	0.049418 ± 0.0022662	0	269.0699	26.907 ± 2.2148	0.047408
0.5	0.5	4	40	250	50	0	506.0344	0.051659 ± 0.0030768	0	265.2644	26.5264 ± 1.613	0.048172
0.5	0.8	4	40	250	50	0	505.2332	0.055459 ± 0.0027238	0	287.2181	28.7218 ± 0.51388	0.051633
0.8	0.2	4	40	250	50	0	504.566	0.051926 ± 0.0037049	0	255.5227	25.5523 ± 1.789	0.048146
0.8	0.5	4	40	250	50	0	507.3963	0.052537 ± 0.0026007	0	260.7711	26.0771 ± 1.1678	0.048688
0.8	0.8	4	40	250	50	0.1	507.0598	0.058388 ± 0.0024248	400	278.129	27.8129 ± 0.73093	0.052559

Tabla A.4-6: Resultados influencia ruido gaussiano, rueda  $\gamma=0.01$

Aplicación de una variación de ruido gaussiano con  $\gamma = 10^{-1}$  a una escena original

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	T.TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0	840.0804	0.029039 ± 0.00077779	0	427.4982	42.7498 ± 1.0697	0.027632
0.2	0.5	4	40	250	50	0	711.7039	0.029589 ± 0.00041778	0	424.103	42.4103 ± 0.38772	0.028585
0.2	0.8	4	40	250	50	0	698.1498	0.031158 ± 0.00072363	0	423.206	42.3206 ± 0.37373	0.030405
0.5	0.2	4	40	250	50	0	840.0815	0.029651 ± 0.00078147	0	415.196	41.5196 ± 0.53452	0.027632
0.5	0.5	4	40	250	50	0	840.2648	0.030236 ± 0.00050467	0	409.2678	40.9268 ± 0.73226	0.029762
0.5	0.8	4	40	250	50	0	840.0846	0.030766 ± 0.00055671	0	413.1581	41.3158 ± 0.42728	0.029596
0.8	0.2	4	40	250	50	0	840.0734	0.029899 ± 0.00020947	0	404.9867	40.4987 ± 0.42869	0.029636
0.8	0.5	4	40	250	50	0	840.083	0.03006 ± 0.00055687	0	400.9345	40.0935 ± 0.50994	0.028717
0.8	0.8	4	40	250	50	0	840.3086	0.031469 ± 0.00030254	0	409.6599	40.966 ± 0.73239	0.031064

Tabla A.4-7: Resultados influencia ruido gaussiano, circuito  $\gamma=0.1$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	T.TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.5	44.5784	0.094064 ± 0.0040986	592	372.6686	37.2669 ± 0.11285	0.089906
0.2	0.5	4	40	250	50	0.3	58.0678	0.095501 ± 0.0040055	506.6667	370.5648	37.0565 ± 0.19321	0.091985
0.2	0.8	4	40	250	50	0.5	56.3706	0.099208 ± 0.0033717	1008	371.3766	37.1377 ± 0.22238	0.094025
0.5	0.2	4	40	250	50	0	57.8814	0.099327 ± 0.0037451	0	359.0571	35.9057 ± 2.1974	0.092763
0.5	0.5	4	40	250	50	0.2	60.0486	0.098654 ± 0.0015644	1220	363.0619	36.3062 ± 0.66538	0.095334
0.5	0.8	4	40	250	50	0.2	63.4884	0.10095 ± 0.0025537	360	363.2698	36.327 ± 0.40818	0.097354
0.8	0.2	4	40	250	50	0.1	48.3709	0.09987 ± 0.0029178	840	358.3752	35.8375 ± 1.6635	0.094742
0.8	0.5	4	40	250	50	0	64.0912	0.099495 ± 0.0018833	0	352.6808	35.2681 ± 0.68158	0.095638
0.8	0.8	4	40	250	50	0.5	43.075	0.10329 ± 0.0015825	824	355.0877	35.5088 ± 0.46237	0.099768

Tabla A.4-8: Resultados influencia ruido gaussiano, pilas  $\gamma=0.1$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	T.TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0	919.6205	0.067815 ± 0.0023872	0	307.8914	30.7891 ± 0.60235	0.063946
0.2	0.5	4	40	250	50	0.2	853.7696	0.071886 ± 0.0023085	1140	307.724	30.7724 ± 0.64651	0.066854
0.2	0.8	4	40	250	50	0	855.8871	0.073885 ± 0.0019894	0	309.9513	30.9951 ± 0.38885	0.070041
0.5	0.2	4	40	250	50	0	918.7621	0.068205 ± 0.0030996	0	289.2554	28.9255 ± 1.583	0.064604
0.5	0.5	4	40	250	50	0.2	919.2267	0.071716 ± 0.0035127	300	293.0812	29.3081 ± 1.0199	0.065559
0.5	0.8	4	40	250	50	0	920.7138	0.074527 ± 0.0023549	0	297.1453	29.7145 ± 0.7607	0.070279
0.8	0.2	4	40	250	50	0	911.1625	0.068886 ± 0.0033322	0	294.21	29.421 ± 1.6329	0.065201
0.8	0.5	4	40	250	50	0	916.365	0.070093 ± 0.0036275	0	284.3208	28.4321 ± 1.246	0.06648
0.8	0.8	4	40	250	50	0.2	43.7294	0.073771 ± 0.0022158	1060	291.7118	29.1712 ± 0.54069	0.071662

Tabla A.4-9: Resultados influencia ruido gaussiano, rueda  $\gamma=0.1$

## A.5: Resultados Influencia del ruido sal y pimienta

### Parámetro Salt&Pepper 0.01

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.9	5.4285	0.010312 ± 0.0050435	551.1111	449.015	44.9015 ± 0.66436	0.0075645
0.2	0.5	4	40	250	50	0.9	2.3379	0.010388 ± 0.0056802	1200	445.4007	44.5401 ± 1.2234	0.0079658
0.2	0.8	4	40	250	50	0.9	7.433	0.012162 ± 0.0062969	608.8889	476.784	47.6784 ± 1.0715	0.0088815
0.5	0.2	4	40	250	50	1	7.6099	0.0085176 ± 0.000342	896	464.9068	46.4907 ± 0.30727	0.0079832
0.5	0.5	4	40	250	50	1	5.0122	0.0087339 ± 0.000253	800	478.2544	47.8254 ± 1.8284	0.0082137
0.5	0.8	4	40	250	50	1	5.6844	0.012748 ± 0.0019236	624	468.0847	46.8085 ± 0.42214	0.0098658
0.8	0.2	4	40	250	50	1	5.8465	0.0091323 ± 0.001123	892	464.7905	46.479 ± 1.1168	0.0080333
0.8	0.5	4	40	250	50	1	12.0265	0.0088911 ± 0.000429	1084	456.9026	45.6903 ± 0.85415	0.0078664
0.8	0.8	4	40	250	50	1	10.1385	0.012282 ± 0.0016766	836	447.9437	44.7944 ± 0.4553	0.010612

Tabla A.5-1: Resultados influencia ruido sal y pimienta, circuito  $\gamma=0.01$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.5	11.6818	0.028589 ± 0.015267	224	376.2301	37.623 ± 0.64354	0.010571
0.2	0.5	4	40	250	50	0.6	10.2323	0.024455 ± 0.014224	773.3333	374.8132	37.4813 ± 0.47534	0.011849
0.2	0.8	4	40	250	50	0.5	22.3464	0.038794 ± 0.014298	640	369.7057	36.9706 ± 0.68591	0.019682
0.5	0.2	4	40	250	50	0.8	4.1189	0.018781 ± 0.011379	855	375.4295	37.5429 ± 0.40041	0.012076
0.5	0.5	4	40	250	50	0.9	9.9859	0.020743 ± 0.010078	937.7778	366.7869	36.6787 ± 0.65649	0.012751
0.5	0.8	4	40	250	50	0.7	18.74	0.031735 ± 0.0076244	874.2857	363.9452	36.3945 ± 0.43061	0.023635
0.8	0.2	4	40	250	50	0.9	14.3456	0.017286 ± 0.011237	933.3333	364.0303	36.403 ± 0.86147	0.013506
0.8	0.5	4	40	250	50	0.9	11.8586	0.02812 ± 0.0099851	702.2222	352.5353	35.2535 ± 0.71623	0.019404
0.8	0.8	4	40	250	50	0.4	9.2837	0.042817 ± 0.010229	370	361.0791	36.1079 ± 0.62141	0.023192

Tabla A.5-2: Resultados influencia ruido sal y pimienta, pilas  $\gamma=0.01$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.3	3.5084	0.032297 ± 0.0092835	586.6667	312.0506	31.2051 ± 0.85109	0.017095
0.2	0.5	4	40	250	50	0.3	7.5092	0.036299 ± 0.0071439	1000	305.5743	30.5574 ± 1.0732	0.026807
0.2	0.8	4	40	250	50	0.6	4.1515	0.03947 ± 0.0076018	640	318.2097	31.821 ± 0.8654	0.027043
0.5	0.2	4	40	250	50	0.4	7.5492	0.029 ± 0.0086473	740	304.2237	30.4224 ± 2.5334	0.018841
0.5	0.5	4	40	250	50	0.7	38.1224	0.036134 ± 0.0059723	714.2857	314.3477	31.4348 ± 1.5196	0.028563
0.5	0.8	4	40	250	50	0.4	33.4906	0.042807 ± 0.0060202	870	315.7594	31.5759 ± 1.0413	0.031508
0.8	0.2	4	40	250	50	0.6	4.4435	0.028454 ± 0.0084215	893.3333	300.4048	30.0405 ± 0.98286	0.018333
0.8	0.5	4	40	250	50	0.5	7.5918	0.039234 ± 0.0060538	816	300.9303	30.093 ± 1.2648	0.028709
0.8	0.8	4	40	250	50	0.6	12.8895	0.039862 ± 0.0088398	893.3333	308.0244	30.8024 ± 0.75477	0.028951

Tabla A.5-3: Resultados influencia ruido sal y pimienta, rueda  $\gamma=0.01$

### Parámetro Salt&Pepper 0.5

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG STD	Best fitness
0.2	0.2	4	40	250	50	0.8	18.4795	0.0181 ± 0.0030046	525	470.6267	47.0627 ± 0.68244	0.01529
0.2	0.5	4	40	250	50	1	20.1371	0.018484 ± 0.0009894	808	463.9756	46.3976 ± 0.56267	0.016395
0.2	0.8	4	40	250	50	1	24.5182	0.019686 ± 0.0008242	900	456.53	45.653 ± 0.47994	0.01799
0.5	0.2	4	40	250	50	0.9	18.903	0.01861 ± 0.00055643	871.11	463.3938	46.3394 ± 0.66555	0.0177
0.5	0.5	4	40	250	50	1	21.5775	0.01884 ± 0.0016686	832	458.1815	45.8181 ± 0.25754	0.016509
0.5	0.8	4	40	250	50	0.9	27.6852	0.021709 ± 0.0007457	662.22	474.2738	47.4274 ± 1.4022	0.02047
0.8	0.2	4	40	250	50	1	16.4485	0.019001 ± 0.0005502	1024	441.6168	44.1617 ± 1.1374	0.018245
0.8	0.5	4	40	250	50	1	17.4429	0.019706 ± 0.001009	876	449.3261	44.9326 ± 0.67096	0.017994
0.8	0.8	4	40	250	50	0.9	17.3872	0.021702 ± 0.0007256	884.44	445.1271	44.5127 ± 0.56381	0.02065

Tabla A.5-4: Resultados influencia ruido sal y pimienta, circuito  $\gamma=0.05$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG STD	Best fitness
0.2	0.2	4	40	250	50	0	53.0676	0.042396 ± 0.0086517	0	373.7228	37.3723 ± 0.17747	0.031889
0.2	0.5	4	40	250	50	0.1	51.5849	0.043414 ± 0.0085623	1520	371.8077	37.1808 ± 0.1665	0.031668
0.2	0.8	4	40	250	50	0.2	57.9134	0.048223 ± 0.0050511	1140	372.4729	37.2473 ± 0.3766	0.041534
0.5	0.2	4	40	250	50	0.2	53.7725	0.044174 ± 0.0086896	1120	374.0696	37.407 ± 0.52144	0.032466
0.5	0.5	4	40	250	50	0	57.2549	0.047997 ± 0.0047609	0	366.6697	36.667 ± 0.53843	0.042596
0.5	0.8	4	40	250	50	0.2	50.0716	0.056214 ± 0.0064432	1020	360.3623	36.0362 ± 0.77949	0.043021
0.8	0.2	4	40	250	50	0.4	53.9532	0.045537 ± 0.007907	520	368.0283	36.8028 ± 1.3417	0.03414
0.8	0.5	4	40	250	50	0.2	57.4509	0.053016 ± 0.0059329	440	349.7806	34.9781 ± 0.64218	0.043135
0.8	0.8	4	40	250	50	0.2	51.6637	0.053922 ± 0.0093281	700	358.9788	35.8979 ± 0.45937	0.038548

Tabla A.5-5: Resultados influencia ruido sal y pimienta, pilas  $\gamma=0.05$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG STD	Best fitness
0.2	0.2	4	40	250	50	0	497.105	0.042263 ± 0.0027622	0	312.4457	31.2446 ± 0.53939	0.038519
0.2	0.5	4	40	250	50	0.1	25.2734	0.042097 ± 0.0032069	640	301.7306	30.1731 ± 1.6574	0.036834
0.2	0.8	4	40	250	50	0.2	357.783	0.048567 ± 0.0016601	760	307.0507	30.7051 ± 0.62779	0.046171
0.5	0.2	4	40	250	50	0	496.652	0.041423 ± 0.0013925	0	255.9959	25.5996 ± 2.1497	0.040022
0.5	0.5	4	40	250	50	0	70.8196	0.045657 ± 0.0018613	0	281.9714	28.1971 ± 1.4535	0.042799
0.5	0.8	4	40	250	50	0.1	514.506	0.050308 ± 0.0018586	920	296.6029	29.6603 ± 0.77952	0.046675
0.8	0.2	4	40	250	50	0	74.0189	0.041994 ± 0.00139	0	257.3622	25.7362 ± 1.9498	0.04086
0.8	0.5	4	40	250	50	0	71.1008	0.046482 ± 0.0021216	0	268.0876	26.8088 ± 1.3379	0.043913
0.8	0.8	4	40	250	50	0	497.105	0.042263 ± 0.0027622	0	312.4457	31.2446 ± 0.53939	0.038519

Tabla A.5-6: Resultados influencia ruido sal y pimienta, rueda  $\gamma=0.05$

### Parámetro Salt&Pepper 0.1

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.8	28.3656	0.023601 ± 0.00070322	850	435.1749	43.5175 ± 1.3744	0.022461

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.5	4	40	250	50	0.8	44.1685	0.025416 ± 0.001616	1170	448.4738	44.8474 ± 0.72129	0.023575
0.2	0.8	4	40	250	50	0.7	24.5699	0.027091 ± 0.0010603	537.1429	446.9089	44.6909 ± 0.36911	0.025943
0.5	0.2	4	40	250	50	0.8	17.7775	0.025759 ± 0.0016599	625	422.375	42.2375 ± 1.866	0.024316
0.5	0.5	4	40	250	50	0.5	736.0319	0.027477 ± 0.0010565	944	414.517	41.4517 ± 0.34114	0.025813
0.5	0.8	4	40	250	50	0.7	25.1695	0.027606 ± 0.00086618	822.8571	419.5494	41.9549 ± 0.16007	0.02651
0.8	0.2	4	40	250	50	0.7	28.6997	0.026037 ± 0.0014477	1194.2857	405.3627	40.5363 ± 0.30421	0.024845
0.8	0.5	4	40	250	50	0.6	23.9351	0.027493 ± 0.00089322	820	406.5536	40.6554 ± 0.55489	0.026038
0.8	0.8	4	40	250	50	0.7	49.3104	0.027983 ± 0.00091382	1057.1429	415.8971	41.5897 ± 0.61678	0.026283

Tabla A.5-7: Resultados influencia ruido sal y pimienta, circuito  $\gamma=0.1$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0	53.9398	0.059918 ± 0.0074424	0	367.7128	36.7713 ± 0.16152	0.051203
0.2	0.5	4	40	250	50	0.1	56.2855	0.062002 ± 0.0053881	1160	368.6671	36.8667 ± 0.067794	0.053311
0.2	0.8	4	40	250	50	0.1	57.5996	0.065994 ± 0.0039351	1440	367.4644	36.7464 ± 0.21147	0.059027
0.5	0.2	4	40	250	50	0	56.3151	0.065616 ± 0.007586	0	362.2552	36.2255 ± 0.26173	0.05553
0.5	0.5	4	40	250	50	0	55.6232	0.065522 ± 0.0061569	0	360.7151	36.0715 ± 0.35178	0.053444
0.5	0.8	4	40	250	50	0.1	65.7486	0.070551 ± 0.0027745	120	359.9954	35.9995 ± 0.75166	0.06596
0.8	0.2	4	40	250	50	0.2	55.0198	0.072903 ± 0.01168	1240	354.5964	35.4596 ± 2.4338	0.055277
0.8	0.5	4	40	250	50	0.3	51.5451	0.068788 ± 0.0062347	626.6667	352.2765	35.2276 ± 0.63472	0.057067
0.8	0.8	4	40	250	50	0	56.5624	0.073059 ± 0.005254	0	356.9344	35.6934 ± 0.53173	0.062007

Tabla A.5-8: Resultados influencia ruido sal y pimienta, pilas  $\gamma=0.1$

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0	832.0188	0.050252 ± 0.0045091	0	303.2061	30.3206 ± 1.7847	0.045091
0.2	0.5	4	40	250	50	0.1	830.0844	0.051269 ± 0.0025057	1160	300.0992	30.0099 ± 0.99828	0.049244
0.2	0.8	4	40	250	50	0.3	31.062	0.056183 ± 0.0026421	973.3333	295.9506	29.5951 ± 1.0697	0.052614
0.5	0.2	4	40	250	50	0	261.621	0.050412 ± 0.0023913	0	272.4487	27.2449 ± 2.163	0.047484
0.5	0.5	4	40	250	50	0	504.7878	0.054674 ± 0.0026922	0	271.7794	27.1779 ± 0.78514	0.050797
0.5	0.8	4	40	250	50	0	499.6321	0.057842 ± 0.0013835	0	287.0531	28.7053 ± 0.78209	0.055009
0.8	0.2	4	40	250	50	0.1	29.069	0.052571 ± 0.0033037	1520	254.8608	25.4861 ± 2.806	0.044639
0.8	0.5	4	40	250	50	0.1	36.231	0.052961 ± 0.0027614	480	262.3583	26.2358 ± 1.708	0.04695
0.8	0.8	4	40	250	50	0.2	506.5711	0.058156 ± 0.0024343	580	278.9598	27.896 ± 0.74461	0.054585

Tabla A.5-9: Resultados influencia ruido sal y pimienta, rueda  $\gamma=0.1$

## A.6: Resultados Influencia del color y decodificación para imágenes RGB

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	1	4.3561	0.0090361 ± 0.0015545	92	979.3058	97.9306 ± 1.6327	0.0079086
0.2	0.5	4	40	250	50	1	5.1237	0.0088263 ± 0.0004433	872	976.466	97.6466 ± 1.9698	0.0081623
0.2	0.8	4	40	250	50	1	4.9826	0.0098598 ± 0.00069304	928	971.366	97.1366 ± 1.3404	0.008517
0.5	0.2	4	40	250	50	1	4.4018	0.0090174 ± 0.00037662	1124	961.7126	96.1713 ± 2.2109	0.0082543
0.5	0.5	4	40	250	50	1	4.3196	0.0092345 ± 0.00033853	1040	946.2356	94.6236 ± 0.20593	0.0084589
0.5	0.8	4	40	250	50	1	5.7824	0.011183 ± 0.0012313	1004	950.305	95.0305 ± 0.33684	0.010064
0.8	0.2	4	40	250	50	1	1.3531	0.0094138 ± 0.00035321	800	941.4035	94.1404 ± 0.2377	0.0087679
0.8	0.5	4	40	250	50	1	5.2741	0.0095222 ± 0.00031111	672	944.0879	94.4088 ± 0.31058	0.0089642
0.8	0.8	4	40	250	50	1	3.7713	0.012765 ± 0.0015829	896	952.7893	95.2789 ± 0.33675	0.010862

Tabla A.6-1: Resultados influencia del color, circuito

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.5	5.0484	0.023591 ± 0.013078	552	787.0775	78.7078 ± 1.1697	0.011315
0.2	0.5	4	40	250	50	0.6	1.3607	0.022195 ± 0.012034	740	786.3184	78.6318 ± 0.84688	0.011909
0.2	0.8	4	40	250	50	0.6	1.6688	0.029872 ± 0.010569	813.3333	781.4569	78.1457 ± 1.0611	0.016733
0.5	0.2	4	40	250	50	0.8	4.9439	0.016075 ± 0.0077554	550	785.1714	78.5171 ± 1.1036	0.01208
0.5	0.5	4	40	250	50	0.8	1.3442	0.025716 ± 0.0098405	905	777.4243	77.7424 ± 0.87744	0.012651
0.5	0.8	4	40	250	50	0.6	4.5876	0.035147 ± 0.0098425	880	767.1906	76.7191 ± 0.70787	0.018738
0.8	0.2	4	40	250	50	1	6.2544	0.013118 ± 0.00042216	820	766.225	76.6225 ± 1.1553	0.012496
0.8	0.5	4	40	250	50	0.7	5.0975	0.030845 ± 0.0093033	651.4286	754.9164	75.4916 ± 1.1322	0.01816
0.8	0.8	4	40	250	50	0.7	27.0867	0.032365 ± 0.0071031	531.4286	768.73	76.873 ± 0.89612	0.024778

Tabla A.6-2: Resultados influencia del color, pilas

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.3	3.8616	0.032847 ± 0.0082577	346.6667	603.9778	60.3978 ± 0.97664	0.014351
0.2	0.5	4	40	250	50	0.5	0.49857	0.026763 ± 0.012034	712	606.9042	60.6904 ± 0.87959	0.014803
0.2	0.8	4	40	250	50	0.6	3.3531	0.036845 ± 0.0091923	1040	595.2544	59.5254 ± 0.63109	0.022557
0.5	0.2	4	40	250	50	0.5	2.6579	0.027752 ± 0.01138	680	599.8754	59.9875 ± 1.5328	0.014866
0.5	0.5	4	40	250	50	0.6	1.254	0.035916 ± 0.006678	726.6667	588.795	58.8795 ± 1.3605	0.02149
0.5	0.8	4	40	250	50	0.7	5.2411	0.037086 ± 0.0078308	777.1429	592.4121	59.2412 ± 0.87281	0.026806
0.8	0.2	4	40	250	50	0.8	1.9894	0.020894 ± 0.0093432	1020	575.4136	57.5414 ± 0.94365	0.015351
0.8	0.5	4	40	250	50	0.6	6.9521	0.038601 ± 0.0067749	666.6667	579.6734	57.9673 ± 0.94502	0.02668
0.8	0.8	4	40	250	50	0.5	6.7564	0.037586 ± 0.0084645	208	589.2881	58.9288 ± 0.86348	0.020665

Tabla A.6-3: Resultados influencia del color, rueda

## A.7: Resultados Combinación de diferentes Operadores LBP

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	1	6.6729	0.0087772 ± 0.001792	264	764.6405	76.4641 ± 0.71722	0.0073664
0.2	0.5	4	40	250	50	1	2.6874	0.009777 ± 0.0018621	1012	758.5231	75.8523 ± 0.14749	0.0081408
0.2	0.8	4	40	250	50	1	11.5221	0.009927 ± 0.0014399	748	765.1175	76.5117 ± 0.54036	0.0084692
0.5	0.2	4	40	250	50	1	7.4957	0.0083431 ± 0.0002739	956	761.4506	76.1451 ± 0.75753	0.0077138
0.5	0.5	4	40	250	50	1	1.2964	0.0086468 ± 0.0001831	836	758.7806	75.8781 ± 0.72385	0.0084505
0.5	0.8	4	40	250	50	1	6.8636	0.010018 ± 0.0008273	720	753.246	75.3246 ± 0.24375	0.0088557

0.8	0.2	4	40	250	50	1	2.7172	0.0086519 ± 0.000312	664	766.4041	76.6404 ± 1.7336	0.0079611
0.8	0.5	4	40	250	50	1	1.7723	0.0090957 ± 0.000986	756	762.0755	76.2075 ± 2.1942	0.00824
0.8	0.8	4	40	250	50	1	8.3683	0.011201 ± 0.0008716	608	756.8646	75.6865 ± 1.2128	0.010263

Tabla A.7-1: Resultados combinación de diferentes Operadores LBP, circuito

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.4	0.98702	0.028905 ± 0.018039	270	635.0258	63.5026 ± 1.0374	0.009678
0.2	0.5	4	40	250	50	0.8	0.93786	0.016356 ± 0.011279	920	629.3261	62.9326 ± 0.94527	0.0099269
0.2	0.8	4	40	250	50	0.4	3.459	0.036058 ± 0.0098327	880	627.689	62.7689 ± 0.63076	0.020124
0.5	0.2	4	40	250	50	0.8	0.64182	0.015603 ± 0.010562	615	617.7487	61.7749 ± 0.79609	0.010336
0.5	0.5	4	40	250	50	0.4	16.5457	0.03316 ± 0.013063	1290	612.6822	61.2682 ± 0.80675	0.018041
0.5	0.8	4	40	250	50	0.4	54.8244	0.036074 ± 0.0080767	910	609.5545	60.9555 ± 0.93657	0.024397
0.8	0.2	4	40	250	50	0.9	2.5909	0.014116 ± 0.0088816	795.55	603.1233	60.3123 ± 0.97088	0.010816
0.8	0.5	4	40	250	50	0.9	1.716	0.018029 ± 0.0067977	688.88	599.5362	59.9536 ± 0.44765	0.01164
0.8	0.8	4	40	250	50	0.6	11.4532	0.032789 ± 0.01262	820	610.6526	61.0653 ± 0.54591	0.016423

Tabla A.7-2: Resultados combinación de diferentes Operadores LBP, pilas

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.4	1.2071	0.029809 ± 0.013934	50	502.9427	50.2943 ± 0.35852	0.013283
0.2	0.5	4	40	250	50	0.8	0.89417	0.021445 ± 0.010453	860	496.403	49.6403 ± 0.87041	0.01373
0.2	0.8	4	40	250	50	0.4	44.5935	0.039718 ± 0.009459	720	497.4558	49.7456 ± 0.35371	0.027536
0.5	0.2	4	40	250	50	0.6	2.7451	0.025356 ± 0.014525	680	490.9331	49.0933 ± 2.427	0.012059
0.5	0.5	4	40	250	50	0.8	2.0827	0.025856 ± 0.0095947	750	491.5915	49.1592 ± 0.51379	0.016049
0.5	0.8	4	40	250	50	0.4	9.9637	0.039477 ± 0.009716	440	495.0592	49.5059 ± 0.29862	0.02372
0.8	0.2	4	40	250	50	0.9	2.69	0.02152 ± 0.011569	600	481.7838	48.1784 ± 0.75747	0.013352
0.8	0.5	4	40	250	50	0.8	5.6565	0.031304 ± 0.0090307	725	477.4242	47.7424 ± 0.73199	0.01969
0.8	0.8	4	40	250	50	0.7	4.8315	0.038518 ± 0.011419	1154.2	488.2509	48.8251 ± 0.76865	0.019508

Tabla A.7-3: Resultados combinación de diferentes Operadores LBP, rueda

## A.8: Resultados Selección de un subconjunto de bins del histograma LBP

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.9	2.1596	0.0011216 ± 0.00075778	448.888	475.8502	47.585 ± 0.2994	0.0073168
0.2	0.5	4	40	250	50	1	2.5073	0.00093481 ± 0.00018352	828	464.9239	46.492 ± 0.0539	0.00079083
0.2	0.8	4	40	250	50	1	1.6757	0.0011522 ± 0.00031837	712	465.1124	46.511 ± 0.2463	0.00090473
0.5	0.2	4	40	250	50	1	3.0518	0.00091068 ± 0.00015215	596	459.4682	45.947 ± 0.1496	0.00079584
0.5	0.5	4	40	250	50	1	3.7618	0.00090064 ± 3.7569e-05	872	456.1292	45.613 ± 0.1798	0.00082865
0.5	0.8	4	40	250	50	1	5.2655	0.0014947 ± 0.00025674	764	455.8189	45.581 ± 0.2252	0.0011883
0.8	0.2	4	40	250	50	1	0.88184	0.00092309 ± 3.7785e-05	936	443.232	44.323 ± 0.2320	0.00086725
0.8	0.5	4	40	250	50	1	1.0035	0.0010264 ± 5.2008e-05	1084	441.432	44.143 ± 0.2077	0.00094402
0.8	0.8	4	40	250	50	1	8.7502	0.0015362 ± 0.0001884	860	444.9153	44.491 ± 0.2306	0.0011933

Tabla A.8-1: Resultados selección de un subconjunto de bins del histograma LBP, circuito

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.3	9.0349	0.0058355 ± 0.0021956	1053.333	400.6563	40.065 ± 0.2984	0.0017437
0.2	0.5	4	40	250	50	0.8	5.4719	0.0032342 ± 0.0021858	850	407.3714	40.737 ± 0.2517	0.0016879
0.2	0.8	4	40	250	50	0.4	48.9628	0.0071033 ± 0.0016364	660	404.9699	40.497 ± 0.2505	0.0046831
0.5	0.2	4	40	250	50	0.8	6.4213	0.0037052 ± 0.0026328	580	400.3726	40.037 ± 0.5385	0.0014807
0.5	0.5	4	40	250	50	1	5.061	0.0041969 ± 0.0025725	928	395.5497	39.555 ± 0.6102	0.0019263
0.5	0.8	4	40	250	50	0.8	26.0726	0.0062013 ± 0.0017175	1010	395.226	39.522 ± 0.5217	0.0037561
0.8	0.2	4	40	250	50	0.8	3.1048	0.0024766 ± 0.0010052	1000	394.3483	39.4348 ± 0.619	0.0016878
0.8	0.5	4	40	250	50	1	24.7709	0.0038529 ± 0.0014289	796	387.9158	38.791 ± 0.4532	0.0024028
0.8	0.8	4	40	250	50	0.4	57.1788	0.0071696 ± 0.0018544	1080	392.0723	39.207 ± 0.7474	0.003836

Tabla A.8-2: Resultados selección de un subconjunto de bins del histograma LBP, pilas

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.4	5.5849	0.0058549 ± 0.0030074	290	340.5257	34.052 ± 0.2560	0.0021472
0.2	0.5	4	40	250	50	0.3	0.77275	0.0057835 ± 0.002218	800	341.0753	34.107 ± 0.4439	0.0022255
0.2	0.8	4	40	250	50	0.6	12.8763	0.0062051 ± 0.0015985	866.6667	339.5095	33.951 ± 0.1993	0.0044289
0.5	0.2	4	40	250	50	0.3	1.3365	0.0058236 ± 0.0026141	960	339.0597	33.906 ± 0.3473	0.0021939
0.5	0.5	4	40	250	50	0.7	12.7562	0.0051311 ± 0.0019432	868.5714	337.6167	33.761 ± 0.3566	0.0027836
0.5	0.8	4	40	250	50	0.5	40.5609	0.0063091 ± 0.0018977	656	336.268	33.626 ± 0.5828	0.0042221
0.8	0.2	4	40	250	50	0.3	4.8106	0.0061724 ± 0.0026772	1240	333.6558	33.365 ± 0.6904	0.002197
0.8	0.5	4	40	250	50	0.7	12.9752	0.0053258 ± 0.0018776	914.2857	331.2803	33.128 ± 0.8313	0.0035871
0.8	0.8	4	40	250	50	0.8	3.2864	0.0061967 ± 0.0018007	705	331.2344	33.123 ± 0.4929	0.0033489

Tabla A.8-3: Resultados selección de un subconjunto de bins del histograma LBP, rueda

## A.9: Resultados Búsquedas Locales

Búsqueda tipo fixed single

F	CR	D	NP	GR	R	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	10	50	1	4.5228	0.007682 ± 0.001485	272	637.8052	63.7805 ± 0.37148	0.0067929
0.2	0.5	4	40	250	10	50	0.9	5.4175	0.010195 ± 0.0064394	977.77	625.3137	62.5314 ± 0.58775	0.0064121



0.2	0.8	4	40	250	10	50	1	2.9882	0.008789 ± 0.001324	700	604.5268	60.4527 ± 1.0945	0.0073114
0.5	0.2	4	40	250	10	50	1	5.0496	0.008931 ± 0.001834	820	589.887	58.9887 ± 0.12775	0.0074183
0.5	0.5	4	40	250	10	50	1	3.6316	0.008190 ± 0.0008692	644	588.4146	58.8415 ± 0.08382	0.0076597
0.5	0.8	4	40	250	10	50	1	5.0241	0.009634 ± 0.0007884	528	592.7044	59.2704 ± 0.16455	0.0083182
0.8	0.2	4	40	250	10	50	1	2.5626	0.0079643 ± 0.000234	736	586.581	58.6581 ± 0.1784	0.0076638
0.8	0.5	4	40	250	10	50	1	3.119	0.0081547 ± 0.000321	728	591.9249	59.1925 ± 0.16991	0.0076396
0.8	0.8	4	40	250	10	50	1	1.5425	0.010957 ± 0.0013102	588	586.1977	58.6198 ± 0.27428	0.0086972

Tabla A.9-1: Resultados búsqueda local fixed single, circuito

F	CR	D	NP	GR	R	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	10	50	0.7	3.4934	0.017413 ± 0.013809	40	517.1915	51.7191 ± 0.68824	0.0086128
0.2	0.5	4	40	250	10	50	0.8	0.69029	0.013727 ± 0.0084569	875	508.2271	50.8227 ± 0.28895	0.0090982
0.2	0.8	4	40	250	10	50	0.6	9.6591	0.025354 ± 0.01178	980	506.1122	50.6112 ± 0.33371	0.013803
0.5	0.2	4	40	250	10	50	0.6	2.4075	0.01832 ± 0.010978	1146.66	498.4055	49.8405 ± 0.31181	0.0098883
0.5	0.5	4	40	250	10	50	0.9	2.1156	0.014366 ± 0.0053592	840	492.5484	49.2548 ± 0.45607	0.010008
0.5	0.8	4	40	250	10	50	0.7	12.8268	0.024048 ± 0.010114	788.57	490.9149	49.0915 ± 0.41834	0.014801
0.8	0.2	4	40	250	10	50	0.8	2.2876	0.014156 ± 0.007311	105	492.6437	49.2644 ± 0.50583	0.01018
0.8	0.5	4	40	250	10	50	1	0.94691	0.018703 ± 0.0054004	732	484.6106	48.4611 ± 0.56201	0.012977
0.8	0.8	4	40	250	10	50	0.4	17.7863	0.032143 ± 0.010608	850	487.2931	48.7293 ± 0.41144	0.018804

Tabla A.9-2: Resultados búsqueda local fixed single, pilas

F	CR	D	NP	GR	R	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	10	50	0.3	1.7273	0.029519 ± 0.012392	40	423.8095	42.381 ± 0.60061	0.011912
0.2	0.5	4	40	250	10	50	0.3	2.1097	0.03138 ± 0.012099	786.66	419.5255	41.9526 ± 0.80757	0.013515
0.2	0.8	4	40	250	10	50	0.4	4.7267	0.036219 ± 0.011085	1130	414.7423	41.4742 ± 0.46703	0.018328
0.5	0.2	4	40	250	10	50	0.6	2.4811	0.022899 ± 0.011673	760	418.4206	41.8421 ± 0.63425	0.013553
0.5	0.5	4	40	250	10	50	0.7	49.2515	0.031847 ± 0.0080838	817.14	412.5385	41.2538 ± 0.8779	0.023262
0.5	0.8	4	40	250	10	50	0.6	11.0269	0.03453 ± 0.0064173	1233.33	412.7377	41.2738 ± 0.78277	0.02724
0.8	0.2	4	40	250	10	50	0.6	0.97336	0.026521 ± 0.01295	786.66	402.9675	40.2968 ± 1.1414	0.012006
0.8	0.5	4	40	250	10	50	0.7	3.2613	0.030391 ± 0.0098524	914.285	400.6874	40.0687 ± 0.77952	0.016243
0.8	0.8	4	40	250	10	50	0.5	44.1494	0.03731 ± 0.006192	1120	409.889	40.9889 ± 0.67621	0.025549

Tabla A.9-3: Resultados búsqueda local fixed single, rueda

## Búsqueda tipo "threshold single"

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	1	1.4366	0.0075709 ± 0.0008612	412	989.3646	98.9365 ± 2.1542	0.0067291
0.2	0.5	4	40	250	50	1	4.9448	0.0079112 ± 0.0011443	1104	983.0019	98.3002 ± 1.3301	0.0069234
0.2	0.8	4	40	250	50	1	0.80275	0.0084247 ± 0.0006602	820	961.2219	96.1222 ± 2.8628	0.0075624
0.5	0.2	4	40	250	50	1	5.0777	0.0077178 ± 0.0003637	876	939.8385	93.9838 ± 3.5469	0.0071233
0.5	0.5	4	40	250	50	1	4.8783	0.0078889 ± 0.0002424	588	938.6816	93.8682 ± 2.3172	0.0075236
0.5	0.8	4	40	250	50	1	3.1779	0.0094062 ± 0.0008832	868	935.4835	93.5483 ± 3.1066	0.0082269
0.8	0.2	4	40	250	50	1	2.0295	0.0081509 ± 0.0002554	1004	874.4438	87.4444 ± 2.4041	0.0077796
0.8	0.5	4	40	250	50	1	5.0543	0.0085578 ± 0.0010048	896	873.239	87.3239 ± 2.9728	0.0078172
0.8	0.8	4	40	250	50	1	6.0521	0.0099929 ± 0.0006723	820	923.1164	92.3116 ± 2.2344	0.0093289

Tabla A.9-4: Resultados búsqueda local threshold single, circuito

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.4	2.2083	0.021659 ± 0.011246	360	838.7887	83.8789 ± 3.0304	0.0084484
0.2	0.5	4	40	250	50	0.6	0.82347	0.018295 ± 0.011082	920	753.1206	75.3121 ± 5.2316	0.0093755
0.2	0.8	4	40	250	50	0.4	4.5922	0.031634 ± 0.013731	680	802.5071	80.2507 ± 2.627	0.010909
0.5	0.2	4	40	250	50	0.8	3.6164	0.013191 ± 0.0069571	1030	744.1122	74.4112 ± 4.4828	0.0091916
0.5	0.5	4	40	250	50	0.9	6.6265	0.014046 ± 0.0050627	942.222	625.5242	62.5524 ± 2.6241	0.010114
0.5	0.8	4	40	250	50	0.7	5.3232	0.021679 ± 0.0077605	1102.85	779.3782	77.9378 ± 1.0641	0.013135
0.8	0.2	4	40	250	50	1	2.8902	0.010756 ± 0.00058831	912	634.7668	63.4767 ± 6.0803	0.01006
0.8	0.5	4	40	250	50	1	4.8105	0.015131 ± 0.002048	704	586.1976	58.6198 ± 1.3484	0.011116
0.8	0.8	4	40	250	50	0.7	8.011	0.021655 ± 0.0073059	765.714	771.7229	77.1723 ± 2.567	0.012256

Tabla A.9-5: Resultados búsqueda local threshold single, pilas

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	0.5	0.91726	0.023646 ± 0.011673	480	694.4657	69.4466 ± 1.3381	0.011538
0.2	0.5	4	40	250	50	0.6	2.0885	0.024731 ± 0.012407	666.666	591.0869	59.1087 ± 3.5302	0.012483
0.2	0.8	4	40	250	50	0.2	44.9555	0.034505 ± 0.0084592	660	689.6499	68.965 ± 0.99451	0.02093
0.5	0.2	4	40	250	50	0.8	1.33	0.019237 ± 0.0092815	825	560.2764	56.0276 ± 4.3263	0.013049
0.5	0.5	4	40	250	50	0.6	6.229	0.02453 ± 0.0087376	806.666	554.3916	55.4392 ± 2.1279	0.014592
0.5	0.8	4	40	250	50	0.5	2.0677	0.033741 ± 0.0080406	992	683.8434	68.3843 ± 1.0251	0.020811
0.8	0.2	4	40	250	50	0.7	0.77155	0.021657 ± 0.010306	994.2857	467.0764	46.7076 ± 5.2254	0.013445
0.8	0.5	4	40	250	50	0.6	3.6913	0.029817 ± 0.0091672	1093.333	545.1075	54.5107 ± 2.1759	0.018384
0.8	0.8	4	40	250	50	0.4	33.3673	0.036132 ± 0.0080025	640	676.6986	67.6699 ± 1.1601	0.024612

Tabla A.9-6: Resultados búsqueda local threshold single, rueda

## Búsqueda tipo "fixed multi"

F	CR	D	NP	GR	SRT	SR	ICD	MBF ± STD	AES	TOTAL TIME	TPR AVG ± STD	Best fitness
0.2	0.2	4	40	250	50	1	5.8908	0.0077519 ± 0.0015588	300	1604.4154	160.4415 ± 1.2172	0.0062141
0.2	0.5	4	40	250	50	1	3.4063	0.0076976 ± 0.0011378	1044	1611.5857	161.1586 ± 1.4452	0.0068562
0.2	0.8	4	40	250	50	1	5.9268	0.0079989 ± 0.00058225	568	1611.59	161.159 ± 1.6009	0.0066506
0.5	0.2	4	40	250	50	1	3.5305	0.0078206 ± 0.00027498	848	1597.5122	159.7512 ± 0.97225	0.0074099
0.5	0.5	4	40	250	50	1	3.5377	0.0081928 ± 0.0010293	1044	1605.2416	160.5242 ± 1.2992	0.0075624
0.5	0.8	4	40	250	50	1	2.3522	0.0092523 ± 0.00075427	852	1603.5535	160.3553 ± 1.1835	0.0081155
0.8	0.2	4	40	250	50	1	2.8248	0.0081106 ± 0.0003627	884	1602.7072	160.2707 ± 1.4408	0.0075604
0.8	0.5	4	40	250	50	1	1.6669	0.0079974 ± 0.00060159	600	1602.842	160.2842 ± 0.84211	0.0069056
0.8	0.8	4	40	250	50	1	3.3761	0.010853 ± 0.0011665	992	1597.5739	159.7574 ± 1.1638	0.0089731

Tabla A.9-7: Resultados búsqueda local fixed multi, circuito



0.8	0.5	3	30	250	18.815	1	9.2	0.0073685 ± 0.00048294	516	126.4797	12.648 ± 0.10489	0.0065042
0.8	0.5	3	30	250	18.815	0.7	15.6961	0.0068708 ± 0.00063757	432.857	131.6858	13.1686 ± 0.04174	0.0056989
0.8	0.5	3	30	250	18.815	0.7	18.2729	0.0071077 ± 0.00061176	394.286	137.2434	13.7243 ± 0.1589	0.0057442
0.8	0.5	3	30	250	18.815	1	7.6703	0.010296 ± 0.0013784	453	116.2149	11.6215 ± 0.059389	0.0082447
0.8	0.5	3	30	250	18.815	1	4.2088	0.0090565 ± 0.0012171	489	114.2804	11.428 ± 0.072788	0.0074908
0.8	0.5	3	30	250	18.815	1	5.366	0.011075 ± 0.00092312	336	116.4056	11.6406 ± 0.058103	0.0096694
0.8	0.5	3	30	250	18.815	1	12.0236	0.0083712 ± 0.00045639	300	119.1729	11.9173 ± 0.013745	0.0074123
0.8	0.5	3	30	250	18.815	0.9	19.0835	0.0078671 ± 0.00062228	470	127.6836	12.7684 ± 0.24801	0.0067653
0.8	0.5	3	30	250	18.815	0.3	19.8713	0.0091532 ± 0.0006537	340	120.9835	12.0984 ± 0.10527	0.0083209
0.8	0.5	3	30	250	18.815	0.9	12.3814	0.0071032 ± 0.00095642	283.333	136.7771	13.6777 ± 0.12645	0.0059703
0.8	0.5	3	30	250	18.815	1	10.0962	0.006289 ± 0.0006169	402	140.0104	14.001 ± 0.1349	0.0053453
0.8	0.5	3	30	250	18.815	0.6	19.9363	0.0077678 ± 0.00075428	350	119.2559	11.9256 ± 0.031334	0.0064983
0.8	0.5	3	30	250	18.815	1	7.2107	0.0078146 ± 0.000652	366	111.6406	11.1641 ± 0.043346	0.0069051
0.8	0.5	3	30	250	18.815	1	14.873	0.0069353 ± 0.0010204	486	124.0516	12.4052 ± 0.06102	0.0052069
0.8	0.5	3	30	250	18.815	1	12.6945	0.0066678 ± 0.00039979	534	132.0233	13.2023 ± 0.15406	0.0061773
0.8	0.5	3	30	250	18.815	1	17.3151	0.0080891 ± 0.00071718	405	125.5774	12.5577 ± 0.059338	0.0070497
0.8	0.5	3	30	250	18.815	1	3.1586	0.0059421 ± 0.00038722	516	125.7525	12.5753 ± 0.028899	0.0055033
0.8	0.5	3	30	250	18.815	1	6.7279	0.018984 ± 0.00036578	465	130.5972	13.0597 ± 0.11626	0.018311
0.8	0.5	3	30	250	18.815	0.9	2.9603	0.0089767 ± 0.00082684	496.666	130.0662	13.0066 ± 0.20438	0.0049027
0.8	0.5	3	30	250	18.815	0	56.959	0.012291 ± 0.00038176	0	128.9246	12.8925 ± 0.024833	0.01176
0.8	0.5	3	30	250	18.815	0.7	5.3617	0.0079818 ± 0.00095732	497.143	128.7633	12.8763 ± 0.025148	0.0064657
0.8	0.5	3	30	250	18.815	1	8.2356	0.010355 ± 0.00043259	432	108.6554	10.8655 ± 0.047634	0.0096577
0.8	0.5	3	30	250	18.815	1	2.6252	0.012668 ± 0.0010798	435	108.5987	10.8599 ± 0.060691	0.010718
0.8	0.5	3	30	250	18.815	1	5.7016	0.0083647 ± 0.00074499	291	118.1399	11.814 ± 0.046729	0.0070351
0.8	0.5	3	30	250	18.815	1	3.2326	0.054644 ± 0.00050818	468	113.3005	11.3301 ± 0.020873	0.054139

Tabla A.10-1: Resultados Retina Disco Óptico

## Anéxo B: Procesos de decodificación detallados

En esta sección se detalla gráficamente las estrategias básicas empleadas para la decodificación según el tipo o modelo de individuo adaptado a la variedad de problema. Se considera que un individuo está decodificado en el momento en el que el genotipo del individuo ha sido transformado a una subimagen a la que le es aplicado el Operador LBP para posteriormente calcular el *fitness* o adecuación de dicho individuo.

Dicho de otro modo, la decodificación en el contexto de este estudio es el proceso de obtener una subimagen de una escena a partir del vector que define al individuo, sobre la cual obtener un vector de características LBP.

### B.1: Consideraciones previas

Es preciso señalar que las coordenadas de una matriz imagen en Matlab se representan del siguiente modo:

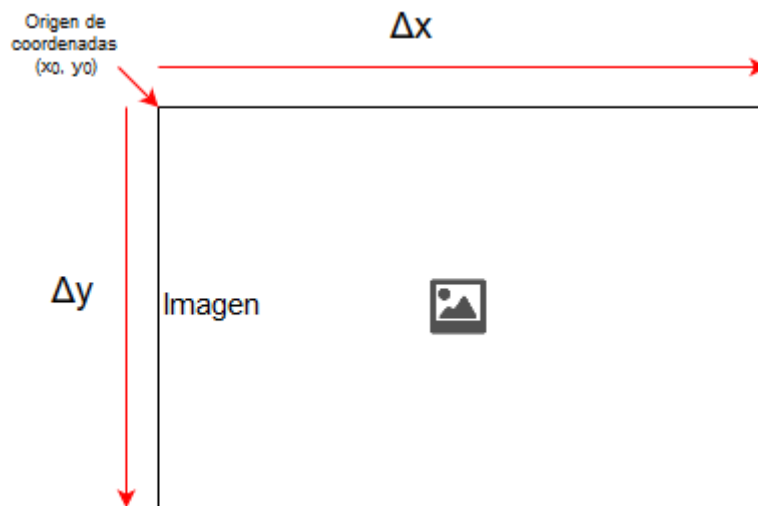


Figura B.1-1: Matriz imagen en Matlab

- El origen de coordenadas se sitúa en la esquina superior izquierda de la matriz imagen
- La anchura, el incremento de la dimensión x, se incrementa en sentido derecho
- La altura, el incremento de la dimensión y, se incrementa hacia abajo

Para todos los casos de decodificación; son dadas una imagen escena y una imagen template:

- $x_0$ : Origen de abscisas del template respecto a la escena
- $y_0$ : Origen de ordenadas del template respecto a la escena
- $\Delta w$ : anchura de la imagen template
- $\Delta h$ : altura de la imagen template

## B.2: Decodificación para el problema básico

Dado un individuo de la forma  $[x, y]$  donde:

- $x$ : Representa la coordenada en el eje de abscisas de la esquina superior izquierda de la ventana subimagen de la escena
- $y$ : Representa la coordenada en el eje de ordenadas de la esquina superior izquierda de la ventana subimagen de la escena

El proceso de decodificación básico sería el siguiente:

### Paso 1

Son cargadas la imagen escena y la imagen template, además es calculado el histograma LBP de la imagen template.

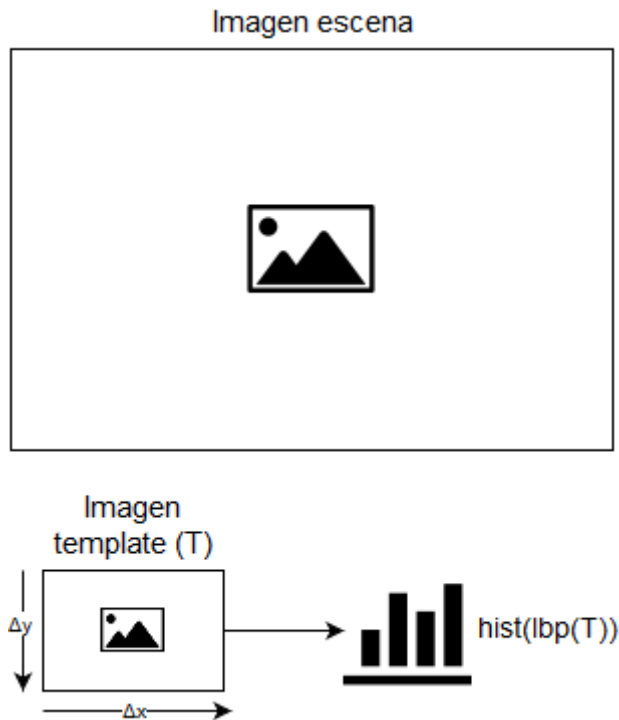


Figura B.2-1: Problema básico, provisión de escena y template

### Paso 2

En el modelo básico de decodificación los individuos están representados por un par de coordenadas  $[x, y]$  respecto a la imagen escena.

Básicamente,  $x$  e  $y$  definen un punto que pertenece a la matriz de la imagen de la escena, tal y como se puede observar en la Figura B.2-2.

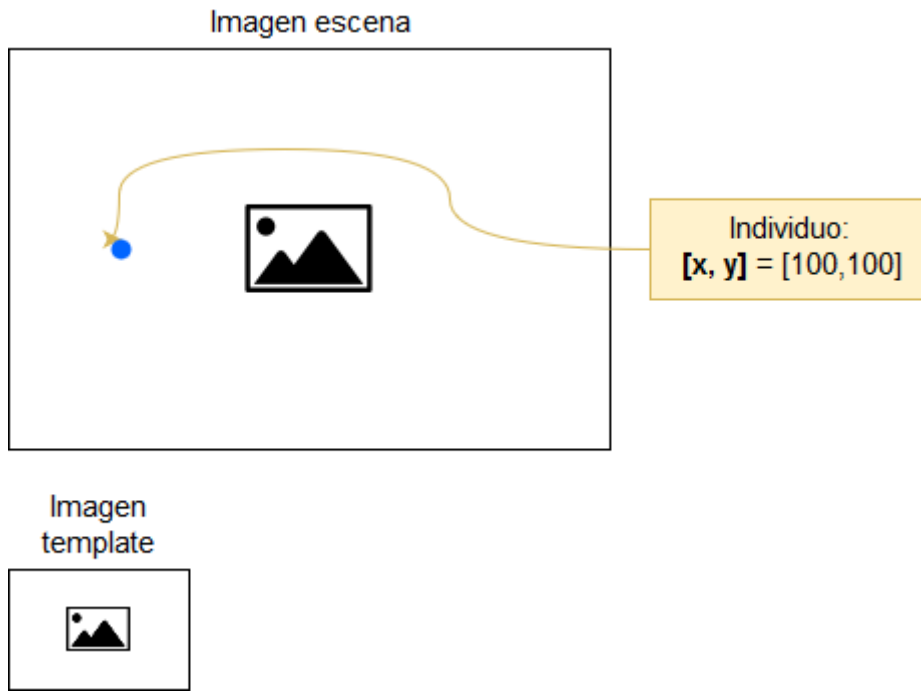


Figura B.2-2: Problema básico, significado de los cromosomas x e y

La altura y la anchura del individuo equivalen a la altura y la anchura del template dado, como se aprecia en la Figura B.2-3. Dichos valores son constantes durante toda la ejecución.

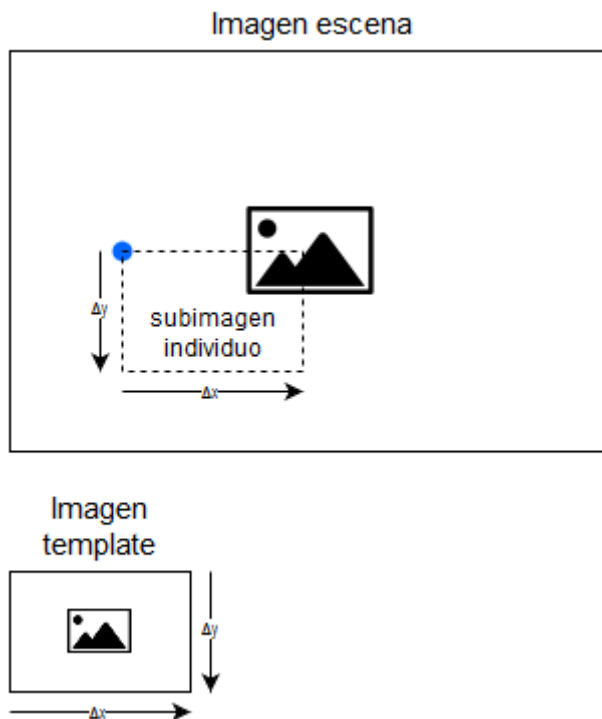


Figura B.2-3: Problema básico, dimensiones del individuo imagen

### Paso 3

Una vez encontrada la matriz imagen del individuo, es calculado su histograma LBP. Cada individuo se decodifica como una subregión de la imagen escena sobre la que calcular su descriptor LBP, como constata la Figura B.2-4.

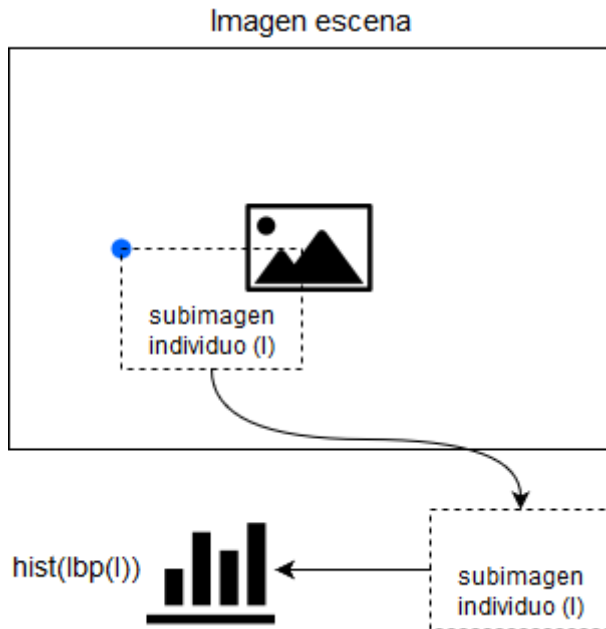


Figura B.2-4: Extracción del individuo imagen y cálculo de su LBP

El histograma LBP del individuo corresponde al fenotipo del mismo. Para asignar un coste a dicho individuo, sencillamente se aplica la función Fitness, que calcula la distancia Euclídea entre su histograma LBP y el histograma LBP del template como en el esquema de la Figura B.2-5.

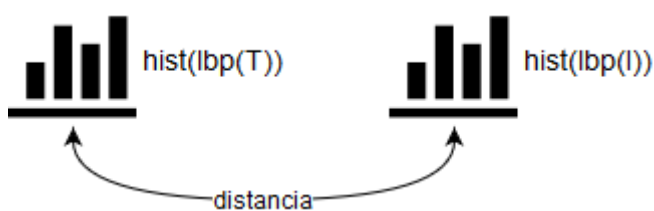


Figura B.2-5: Problema básico, asignación de fitness

### B.3: Decodificación para el problema de template escalado

Dado un individuo de la forma  $[x, y, \alpha]$  donde:

- $x$ : Representa la coordenada en el eje de abscisas de la esquina superior izquierda de la ventana subimagen de la escena
- $y$ : Representa la coordenada en el eje de ordenadas de la esquina superior izquierda de la ventana subimagen de la escena
- $\alpha$ : Representa un factor de cambio de escala

El proceso de decodificación considerando el factor de escalado sería el siguiente:

### Paso 1

Son cargadas las imágenes escena y template, Figura B.3-1. Este template ha sufrido previamente una transformación de cambio de escala. Los límites del factor de escalado se sitúan en el intervalo  $[0.5, 2.0]$ .

Se calcula el histograma LBP de la imagen template y son almacenadas las dimensiones originales del template; serán empleadas posteriormente en la decodificación.

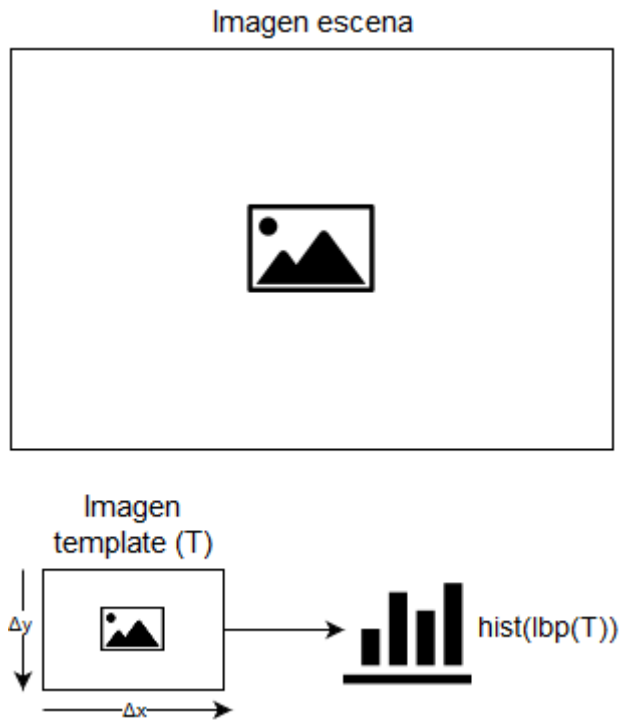


Figura B.3-1: Problema con cambio de escala, provisión de escena y template

### Paso 2

Para el modelo que incluye factor de escalado, los individuos se representan añadiendo una dimensión adicional;  $[x, y, \alpha]$ . Al igual que en el modelo básico, la coordenada de la esquina superior izquierda de la imagen correspondiente al individuo se sitúa directamente donde indican los cromosomas  $x$  e  $y$ , como indica la Figura B.3-2.



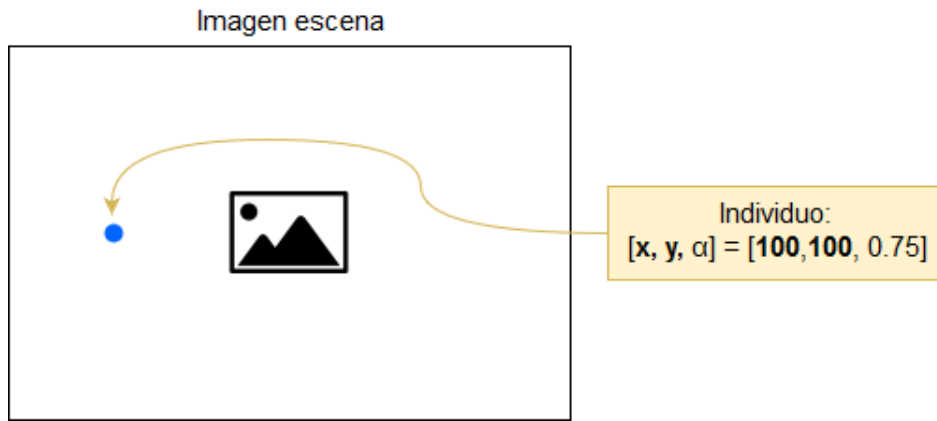


Figura B.3-2: Problema con cambio de escala, significado de los cromosomas x e y

La subimagen del individuo extraída de la escena, Figura B.3-3, es de dimensión  $\alpha(\Delta x * \Delta y)$ , donde  $\Delta x$  y  $\Delta y$  son las dimensiones del template y  $\alpha$  el factor de escalado del individuo

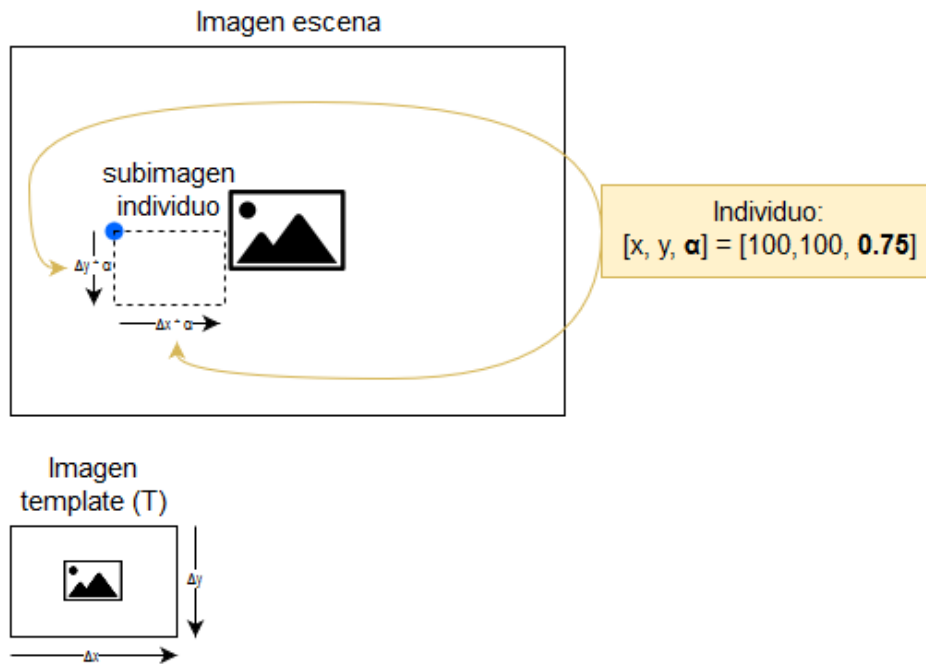


Figura B.3-3: Problema con cambio de escala, dimensiones del individuo imagen

Una vez definida la matriz imagen del individuo que es extraída de la escena, Figura B.3-4, se le vuelve a aplicar el factor de escalado, de forma inversa, para recuperar el tamaño original. Posteriormente el histograma LBP de la subimagen es calculado.

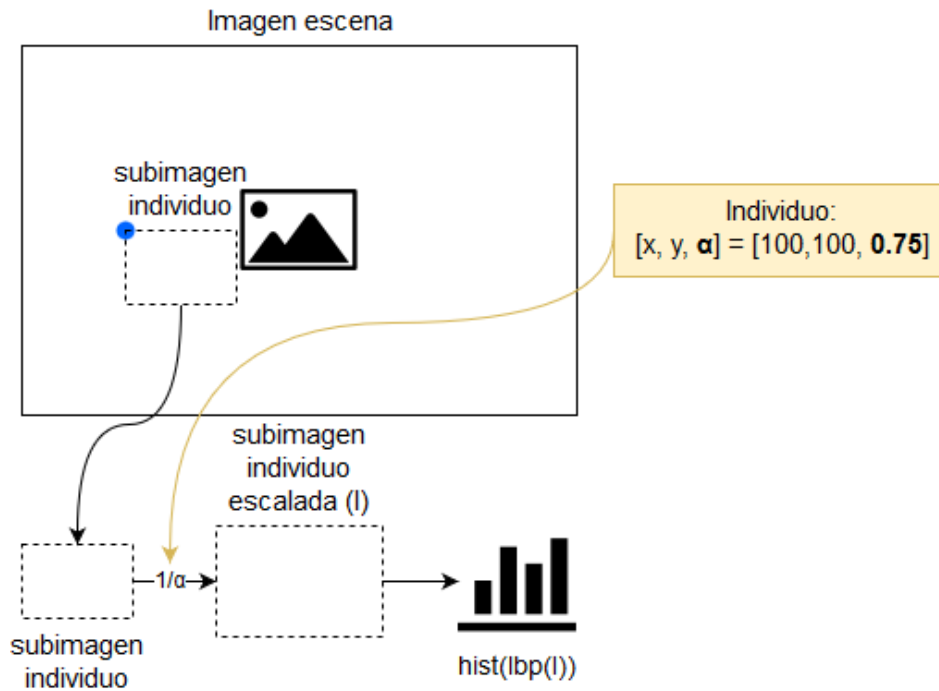


Figura B.3-4: Problema con cambio de escala, extracción del individuo imagen y cálculo de su LBP

### Paso 3

Finalmente se calcula la distancia Euclídea entre el histograma LBP del individuo decodificado y el histograma LBP del template.

## B.4: Decodificación para el problema con rotación

Dado un individuo de la forma  $[x, y, \theta]$  donde:

- $x$ : Representa la coordenada en el eje de abscisas de la esquina superior izquierda de la ventana subimagen de la escena
- $y$ : Representa la coordenada en el eje de ordenadas de la esquina superior izquierda de la ventana subimagen de la escena
- $\theta$ : Representa un ángulo de rotación

El proceso de decodificación considerando el ángulo de rotación sería el siguiente:

### Paso 1

Carga de las imágenes escena y template, Figura B.4-1, cálculo del histograma LBP del template, y también se almacena el valor de la diagonal ( $d$ ) de la imagen template.

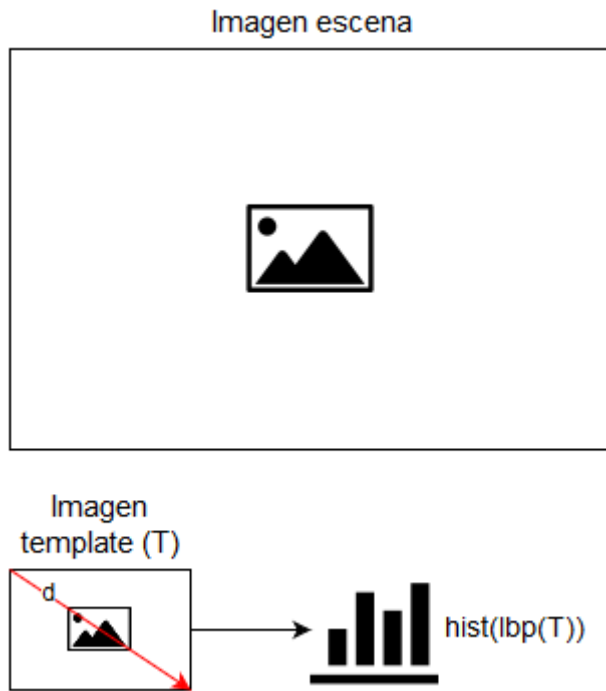


Figura B.4-1: Problema con rotación, provisión de escena y template

## Paso 2

Para la extracción del individuo, para los casos donde se contempla un factor de rotación existen algunos aspectos a tener en cuenta. Los cromosomas  $x$  e  $y$  del individuo, Figura B.4-2, no corresponden con la esquina superior izquierda del individuo decodificado.

Aplicar una transformación de rotación sobre una matriz imagen, implica que la imagen transformada además de la rotación también sufre una variación de tamaño. Por esta razón, los cromosomas  $x$  e  $y$  en los casos de rotación identifican un punto de extracción que no concuerdan con las coordenadas de la esquina superior izquierda de la subimagen a la que será aplicado el operador LBP: será necesario aplicar algunas operaciones adicionales para evitar incorporar fondo extra a la subimagen individuo.

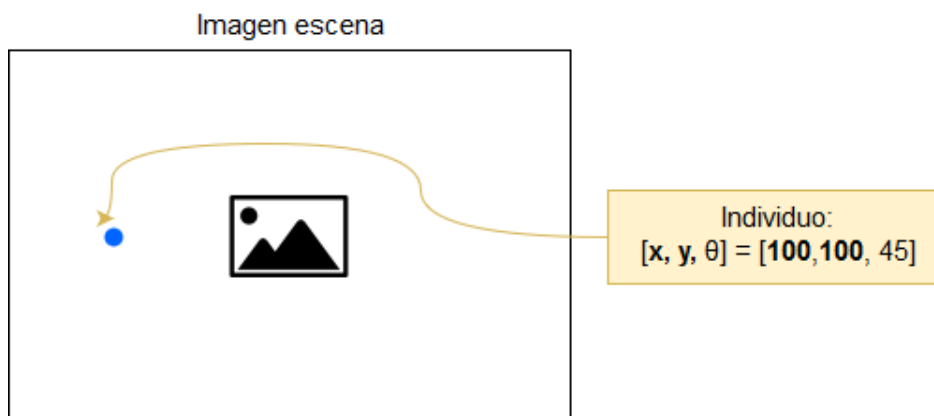


Figura B.4-2: Problema con rotación, significado de los cromosomas  $x$  e  $y$

Inicialmente para decodificar al individuo, es extraída una primera subimagen de la escena, Figura B.4-3, teniendo en cuenta las coordenadas definidas por los cromosomas x e y. La anchura y altura de la subimagen extraída coinciden en longitud con el valor de la diagonal de la imagen template.

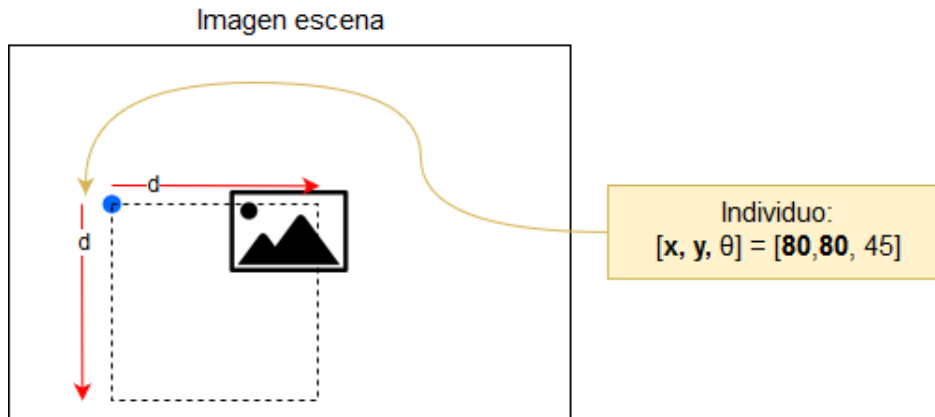


Figura B.4-3: Problema con rotación, dimensiones del individuo imagen

Una vez que ha sido extraída la subimagen de la escena, se le aplica a la misma la transformación de rotación, Figura B.4-4, de acuerdo al valor del tercer cromosoma.

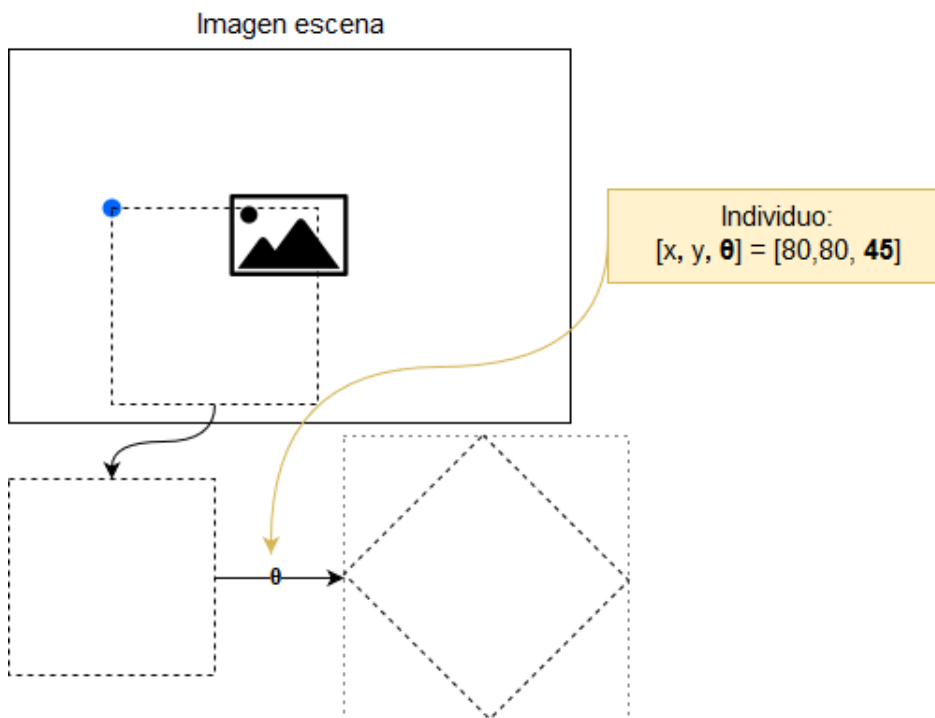


Figura B.4-4: Problema con rotación, extracción 1

Tras rotar la subimagen, se extrae una subimagen más, Figura B.4-5: su tamaño coincide en altura y anchura con los del template dado y se extrae a partir de un punto calculado en base al tamaño de la subimagen primaria rotada y el tamaño del template.

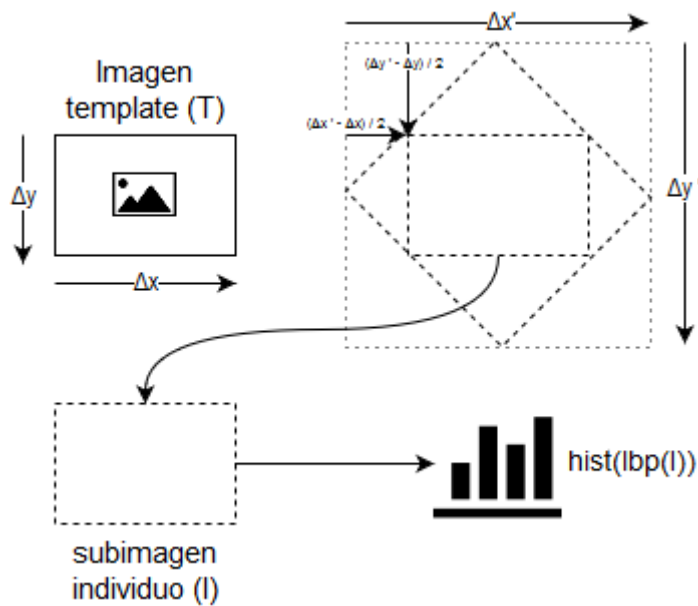


Figura B.4-5: Problema con rotación, extracción 2

### Paso 3

Es calculada la distancia entre los histogramas LBP y asignado dicho valor como cose del individuo.

### B.5: Decodificación para el problema del template escalado y rotado

Dado un individuo de la forma  $[x, y, \alpha, \theta]$  donde:

- $x$ : Representa la coordenada en el eje de abscisas de la esquina superior izquierda de la ventana subimagen de la escena
- $y$ : Representa la coordenada en el eje de ordenadas de la esquina superior izquierda de la ventana subimagen de la escena
- $\alpha$ : Representa un factor de cambio de escala
- $\theta$ : Representa un ángulo de rotación

El proceso de decodificación considerando factor de escalado y ángulo de rotación sería el siguiente.

## Paso 1

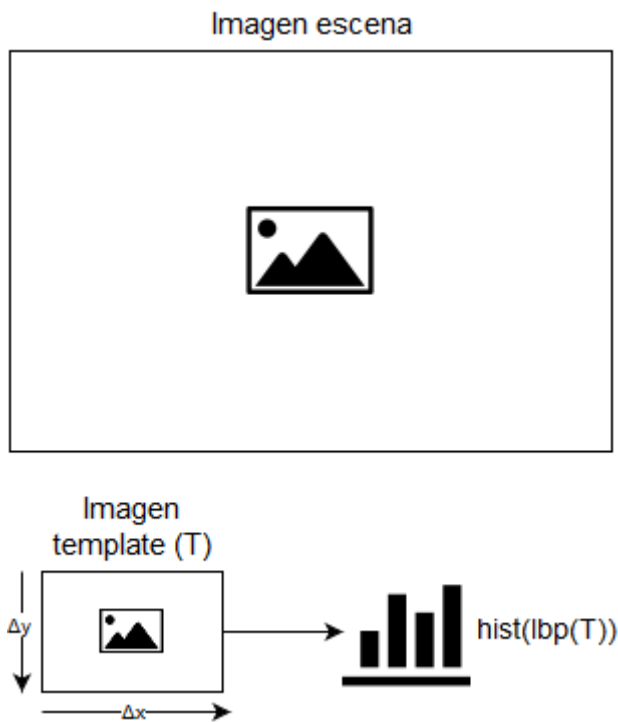


Figura B.5-1: Problema con escalado y rotación, provisión de escena y template

Inicialmente han sido cargados una escena y un template, Figura B.5-1. Dicho template incluye dos transformaciones, una de cambio de escala y otra de rotación.

## Paso 2

En primer lugar, las coordenadas  $x$  e  $y$ , las dos primeras dimensiones del individuo se emplean como punto de referencia en la escena según la Figura B.5-2.

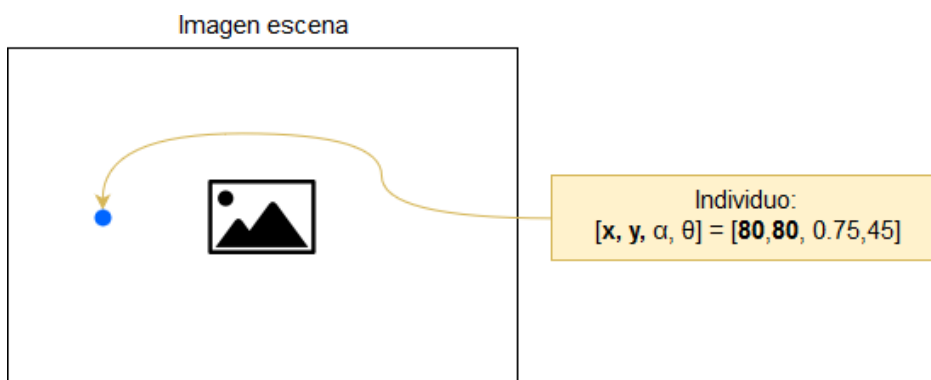


Figura B.5-2: Problema con escalado y rotación, significado de los cromosomas  $x$  e  $y$

A partir de las coordenadas  $x$  e  $y$ , se toman como longitud y altura para extraer una subimagen de la escena, Figura B.5-3, la diagonal resultante de aplicar la dimensión de escalado a la anchura y altura del template.

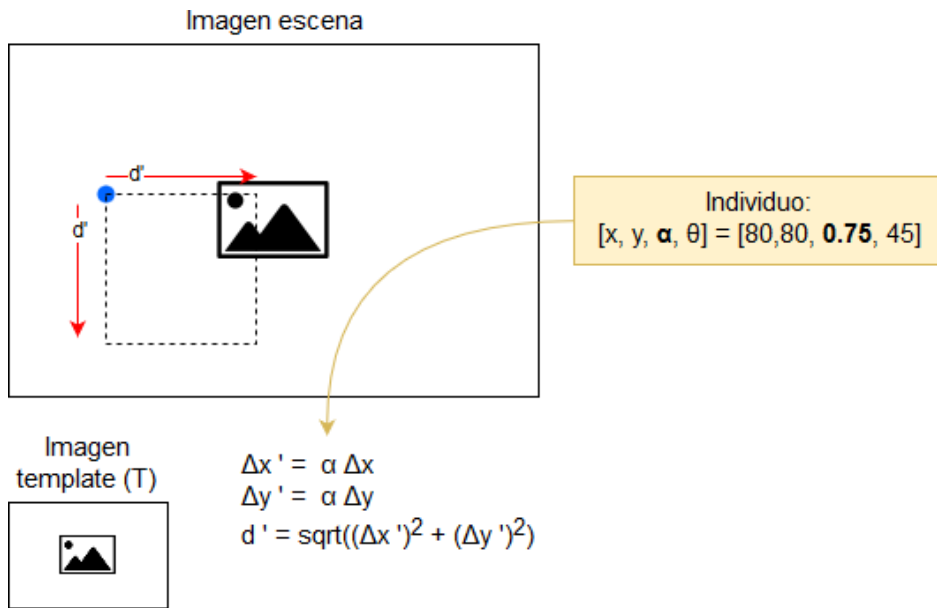


Figura B.5-3: Problema con rotación y escalado, dimensiones de extracción

Tras extraer esta primera subimagen, se rota aplicando el valor que define la dimensión de rotación del individuo como indica la Figura B.5-4.

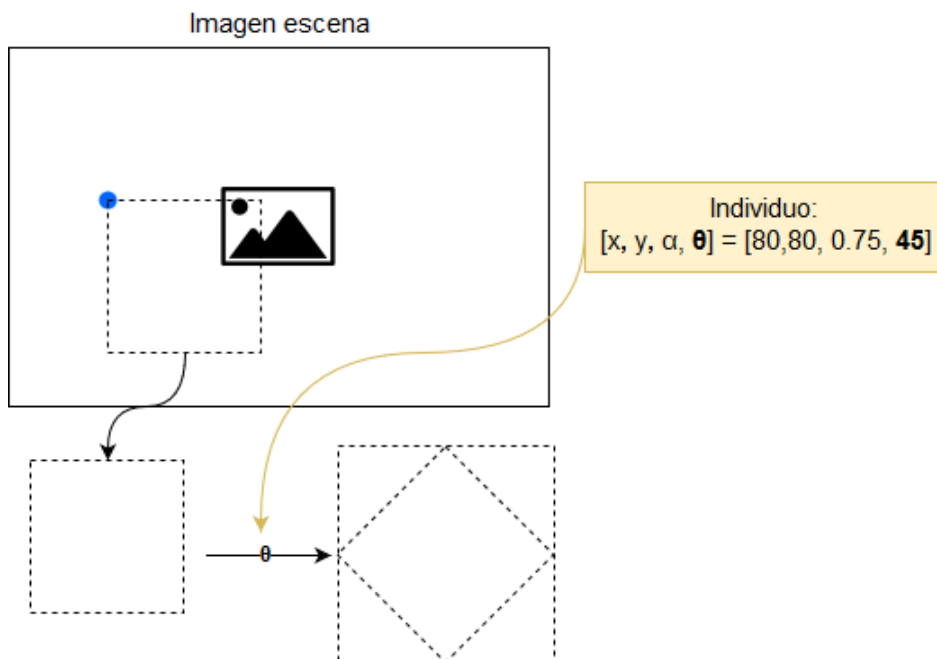


Figura B.5-4: Problema con escalado y rotación, significado del cromosoma  $\theta$

Posteriormente se extrae una nueva subimagen, Figura B.5-5, de la imagen extraída previamente; para evitar incorporar falso fondo. Una vez realizada esta operación, se aplica la inversa del factor de escalado definido por el cromosoma del individuo.

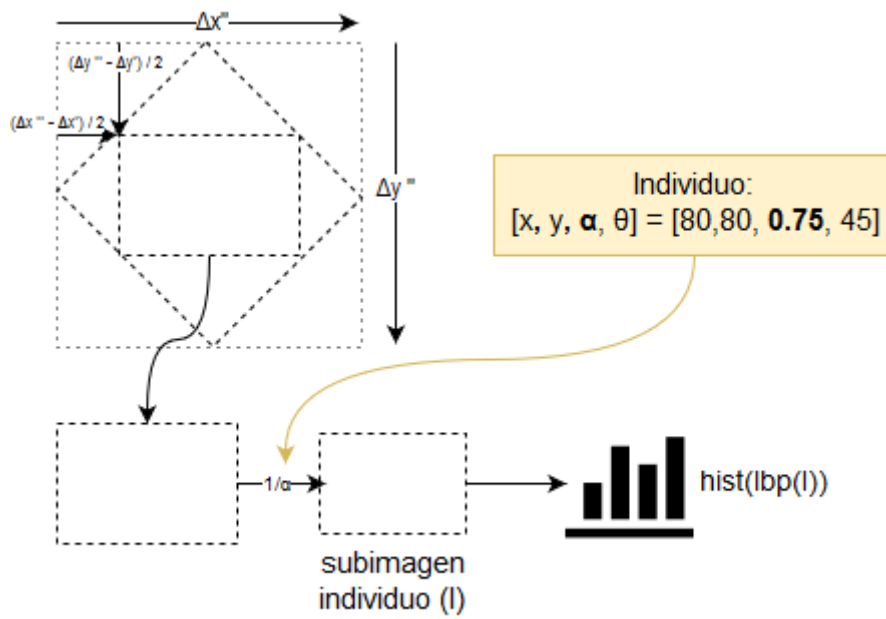


Figura B.5-5: Problema con escalado y rotación, significado del cromosoma  $\alpha$

### Paso 3

Es calculado el histograma LBP.



## **Anexo C: Visión Artificial y Computación Evolutiva**

Las disciplinas científicas y académicas de la Visión Artificial y los Algoritmos Evolutivos son dos campos de especial interés en el mundo de la investigación. Ambas pertenecen al mundo de la Inteligencia Artificial.

La Visión Artificial porque a pesar de los avances que se han logrado en las últimas décadas sigue siendo una materia donde abundan misterios y problemas aún por resolver. Los Algoritmos Evolutivos, una gran familia de algoritmos y procesos de optimización estocástica, a pesar de tratarse de una materia de clásica estudio, siguen siendo una potente herramienta que emplear a la hora de construir metodologías para tratar de resolver nuevos problemas debido a la gran aplicabilidad que manifiestan ante distintas situaciones para conducir procesos de optimización.

### **Visión Artificial**

La toma de decisiones en base a percepciones del entorno, es una característica propia de los seres vivos que la Inteligencia Artificial lleva investigando desde hace décadas con el fin de modelizar sistemas inteligentes capaces de reaccionar ante un entorno cambiante así como para contribuir a la comprensión de naturaleza de los seres vivos y de la mente humana con los hallazgos que se han ido aportando durante este tiempo.

La percepción visual es sin duda el sentido más desarrollado de los seres humanos, por lo que en cuanto fue posible conectar una cámara a un microprocesador, evento que ocurrió a finales de la década de los años 50 del siglo XX y principios de la década de los años 60, comenzaron las primeras investigaciones orientadas al tratamiento automático de imágenes donde claramente destacan los estudios de Lawrence G. Roberts como uno de los pioneros en asentar las bases de la nueva disciplina, con trabajos como su propia tesis doctoral (Roberts, 1963), donde es descrito e implementado un método para que un computador modele un modelo tridimensional a partir de una imagen, partiendo de unas suposiciones previas de la imagen.

Inicialmente, los expertos consideraron que sería una tarea sencilla dado que los seres humanos son excepcionalmente eficaces interpretando imágenes, pero se equivocaron y pronto comenzaron a descubrir los numerosos retos adyacentes a la interpretación sistemática de imágenes.

A partir de tal momento puede decirse que la Visión Artificial estaba constituida como disciplina y desde entonces se han realizado numerosos avances desde distintos enfoques (Learned-Miller, 2011). Inicialmente, dichos avances se produjeron de manera discreta, hoy en día la Visión Artificial es un campo muy activo con una notable madurez pero

todavía se enfrenta a grandes retos y aún está lejos de acercarse a la capacidad humana de interpretación de escenas.

### **Campos de aplicación**

Dado el contexto de esta lectura, el más evidente ámbito de aplicación de la Visión Artificial es la investigación en Inteligencia Artificial. En ocasiones, se considera que la Visión Artificial en conjunto con la Robótica constituye una disciplina convergente denominada *Machine Vision* (Sonka, Hlavac, & Boyle, 2014). No obstante éste área no dejaría de estar incluida en la disciplina de la Inteligencia Artificial.

Más allá de la investigación científica, la Visión Artificial tiene aplicabilidad en numerosas actividades humanas, especialmente en la medicina, la industria y el comercio, o la defensa, por citar algunas de estas aplicaciones: las posibilidades son ilimitadas.

El campo de la medicina supone un nicho de grandes posibilidades y oportunidades a los investigadores en Visión Artificial para desarrollar o perfeccionar métodos y algoritmos de diagnóstico automático en base al procesamiento masivo y sistemático de imágenes digitales clínicas, como radiografías, imágenes de resonancia magnética o imágenes de retina. Este enfoque permite agilizar la labor de los médicos para realizar diagnósticos y otras actividades.

Dentro de este nicho de la medicina, podemos destacar estudios más concretos como (Carmona, Rincón, García-Feijóo, & Martínez-de-la-Casa, 2008), donde se describe la implementación de un sistema de Visión Artificial destinado a la localización de la cabeza del nervio óptico en imágenes de retina.

En (Galbally, Ross, Gomez-Barrero, Fierrez, & Ortega-Garcia, 2013) se plantean mecanismos para construir imágenes sintéticas del iris a partir de templates con codificación binaria.

Un recopilatorio sobre estudios relativos al análisis de imágenes clínicas en general, basados en algoritmos y métodos de Visión Artificial puede consultarse en (Viergever, y otros, 2016).

En el mundo de la industria, la supervisión de los procesos de manufacturación y de los procesos de control de calidad es uno de los roles donde la Visión Artificial goza de un protagonismo especial, una revisión sobre las numerosas aplicaciones de la Visión Artificial en procesos industriales puede encontrarse en (Malamas, Petrakis, Zervakis, Petit, & Legat, 2003).

A nivel comercial, el rango de posibles aplicaciones es igualmente amplio, la detección de eventos en base al control de movimiento (Malamas, Petrakis, Zervakis, Petit, & Legat, 2003), la interacción con humanos así como la clasificación de las emociones de estos

(Busso, y otros, 2004), o control de plagas (Patil & Kumar, 2011) serían algunos recopilatorios donde se analizan estudios que podrían categorizarse con una cierta finalidad comercial.

En defensa, la navegación autónoma de vehículos aéreos no tripulados, la asignación automática de blancos u objetivos y el control fronteras o vigilancia de instalaciones (Mira, Martínez, Rincón, Bachiller, & Fernández-Caballero, 2007), son actividades donde la Visión Artificial ha tenido un creciente interés las últimas décadas.

Merece una mención especial el área de la Realidad Aumentada, ya que el papel de la aplicación de métodos tradicionales de Visión Artificial es crucial. Puede consultarse (Billingham, Clark, Lee, & others, 2015) para adquirir una noción general del estado de los avances recientes concernientes a este área.

Como se manifestado al inicio de esta revisión de las aplicaciones de la Visión Artificial, esta disciplina puede aplicarse en infinidad de actividades humanas, enumerar todas estas aplicaciones es una tarea compleja, por lo que solo se ha ofrecido al lector algunos ejemplos y estudios de referencia.

## **Niveles**

La disciplina de Visión Artificial puede encontrarse organizada por niveles en la literatura académica. Dichos niveles están relacionados con el tipo de actividades que se ejecutan durante el proceso de tratamiento de una imagen digital, desde su adquisición hasta su interpretación. Distintas actividades propias de la Visión Artificial, no necesariamente tienen por qué ser exclusivas de cada nivel, pero ciertas actividades frecuentemente intervienen más a menudo en unos niveles que en otros.

### **Nivel Bajo**

La adquisición de la imagen y la preparación de la misma ejecutando acciones sobre ella como la eliminación de ruido para facilitar los procesamientos posteriores, son tareas de la Visión Artificial que ocupan este nivel en numerosas ocasiones, Otras tareas, como la segmentación, que es la descomposición de imágenes en distintas regiones según un determinado criterio también es una actividad que recurrentemente es considerada como de bajo nivel.

Existen numerosas aproximaciones para la preparación inicial de las imágenes, tanto para eliminación de ruido o artefactos como para la segmentación. Una reciente publicación sobre técnicas de segmentación está disponible en (Khan, 2014).

Los filtros para la detección de bordes u otros fines, la umbralizaciones basadas en intensidad de niveles de gris o el estudio del flujo óptico para imágenes secuenciales, serían algunos de tipos de segmentación.

En esencia, este nivel se destina a aplicar transformaciones sobre la imagen adquirida como fase de preparación necesaria para pasos posteriores de procesamiento. Dicha preparación seguirá una aproximación u otra dependiendo de cuál sea el objetivo final del análisis de la imagen.

### **Nivel Medio**

Podría decirse que las operaciones a realizar sobre una imagen que incluyen conocimiento adicional, como podría ser la inclusión de conocimiento geométrico o la aplicación de segmentación basada en conocimiento, son consideradas tareas de Visión Artificial de Nivel Medio.

Un ejemplo simple de una labor del segundo nivel, sería etiquetar regiones segmentadas, en base a su parecido con figuras geométricas.

La finalidad de las actividades conducidas en el segundo Nivel es darle un primer significado a las distintas regiones segmentadas de la imagen. Dicho de otro modo, en este nivel tendrían labor las labores de clasificación iniciales.

### **Nivel Alto**

Reconocer de forma sistemática los objetos o las actividades que intervienen en una imagen es el objetivo fundamental de la Visión Artificial. Una vez realizadas las etapas de los niveles anteriores, es necesario introducir conocimiento del dominio para lograr interpretar la escena de forma satisfactoria.

Esto implica incorporar procesos de razonamiento y/o descomponer el problema en distintos niveles de descripción. Es decir, para la correcta interpretación de la imagen se recurre a modelar la información extraída de las etapas anteriores en sistemas de representación que faciliten la interpretación.

## **Computación Evolutiva**

Modelar en un sistema computacional la Teoría de la Evolución consiste en diseñar algoritmos que imitan la capacidad de una población de organismos vivos para adaptarse a su entorno con mecanismos de selección mutación y reproducción. En los últimos cuarenta años, diversos métodos de optimización estocástica se han basado en este principio.

La Computación Evolutiva (Eiben & Smith, 2003) o Algoritmos Evolutivos es un nombre común a estas técnicas, entre las cuales el lector puede estar más familiarizado con algoritmos genéticos, estrategias evolutivas o programación genética.

Los rasgos comunes de estas técnicas son modelar poblaciones cuyos miembros representan puntos en un espacio de búsqueda que evolucionan bajo la acción de

operadores estocásticos. La evolución suele organizarse en generaciones con descendientes simplificando la genética natural.

Los elementos que intervienen en la evolución, denominados operadores evolutivos, son la selección de los miembros de la generación actual para producir nuevos miembros en base a un operador genético denominado cruce, la mutación de algunos elementos de la descendencia y de nuevo una selección para determinar qué miembros constituirán la siguiente generación.

La eficiencia de un algoritmo evolutivo depende en gran medida de la parametrización: las poblaciones sucesivas (generaciones) tienen que converger hacia la solución, es decir, a menudo el óptimo global de una función de coste. Una gran parte de la investigación teórica sobre algoritmos evolutivos se dedica al delicado problema de la convergencia— en el sentido de que la búsqueda se dirija al óptimo global y no caiga en óptimos locales— así como a intentar averiguar qué problema es fácil o difícil de resolver para un algoritmo evolutivo

### **Familias de la Computación Evolutiva**

Existen distintas familias de algoritmos evolutivos en la Computación Evolutiva. Se diferencian fundamentalmente en base a la representación de los individuos de la población y en cómo y cuándo actúan los operadores genéticos para producir nuevas generaciones.

Los Algoritmos Genéticos (Holland, 1975) son la familia o variante quizá más popular debido que su implementación es sencilla. Los individuos de la población se representan mediante un array binario, es decir su genotipo es una cadena de bits donde cada gen es un bit. El operador recombinación, que tiene distintas variantes, en esencia utiliza dos miembros de la población actual y recombina sus genes para formar individuos nuevos. El operador mutación recorre el array de cada nuevo individuo y en base a una probabilidad muta el bit del gen actual. La decodificación para obtener fenotipos es dependiente del problema, así como la asignación de un coste. El operador selección determina que miembros pasarán a la siguiente generación.

Las Estrategias Evolutivas (Rechenberg, 1972) persiguen resolver problemas complejos de optimización en el dominio de los números reales. Por lo tanto, los genotipos de los individuos de la población, su representación, consisten en arrays de números reales. La selección de los individuos que crearán descendencia se escogen aleatoriamente de la población, destacando que los descendientes pueden tener más de un padre dependiendo del tipo de selección parental empleada. Si bien la recombinación puede ser similar a la de los Algoritmos Genéticos, la mutación consiste en aplicar una perturbación gaussiana al genotipo, basada en ocasiones en valores de media y desviación que también son

cromosomas del individuo. Los supervivientes pueden elegirse o bien solo de la descendencia creada durante la recombinación, o bien en base la unión de la descendencia y la población actual.

La Programación Genética (Koza, 1992) cambia la forma de representación por completo, en este paradigma los genotipos son árboles, grafos acíclicos no dirigidos, donde cada gen es un nodo del grafo. La selección parental es proporcional al fitness. El operador recombinación efectúa intercambios entre subárboles de los padres, la mutación en base a una probabilidad realiza un cambio aleatorio en un nodo. La generación recientemente creada suele reemplazar a la generación anterior como método de selección de supervivientes.

En la familia de la Evolución Gramatical (Ryan, Collins, & Neill, 1998) los individuos, sus genotipos, se representan mediante números enteros, pero estos son decodificados de acuerdo a una gramática regular en la forma de Backus Naur. La selección de progenitores se efectúa por torneo, seleccionando aleatoriamente los participantes del torneo de la población actual. La recombinación al igual que en los Algoritmos Genéticos, se efectúa intercambiando genes (en este contexto, codones). La mutación consiste en la duplicación o eliminación probabilística de codones, lo que implica que los genotipos de los distintos miembros de la población no tienen por qué tener el mismo tamaño. La supervivencia puede consistir en un reemplazo completo de la generación anterior, o en la selección de los individuos con más adecuación de la unión de la población actual y la descendencia.

La Evolución Diferencial, Differential Evolution en inglés, (Storn & Price, 1997) es otro miembro de familia de la Computación Evolutiva. Se basa fundamentalmente en las aproximaciones de los Algoritmos Genéticos y las Estrategias Evolutivas.

Cabe mencionar que existen otros algoritmos también basados en poblaciones que no son miembros de la familia de los Algoritmos Evolutivos como por ejemplo los algoritmos *Ant Bee Colony* o *Particle Swarm Optimization* (Kennedy, 2006).

Tanto los algoritmos de la Computación Evolutiva, así como los mencionados en el párrafo anterior pueden considerarse Metaheurísticas dada su gran aplicabilidad a distintos tipos de problema.