



UNIVERSIDAD NACIONAL DEL CENTRO DE LA PROVINCIA DE BUENOS AIRES

AUTÓMATAS DE LATTICE-BOLTZMANN PARA MODELOS DE ILUMINACIÓN DIFUSA APLICADOS A LA DETECCIÓN DE TEXTURAS EN IMÁGENES DIGITALES

Tesis de Grado en Ingeniería de Sistemas

Por

Emmanuel Iarussi

Directores

Dr. Alejandro Clausse

Mg. María Virginia Cifuentes

"Caminando en línea recta no puede uno llegar muy lejos."

El Principito, Antoine de Saint-Exupéry, 1943

Resumen

Se propone una técnica para la detección y segmentación de texturas en imágenes digitales. Esta técnica está basada en lattice-Boltzmann, un método ampliamente difundido en la simulación de fluidos que ha sido recientemente utilizada en la creación de modelos de iluminación. A partir de la simulación de la dispersión lumínica en medios generados utilizando las imágenes que se pretenden procesar, y mediante el estudio de observables físicos generados desde la simulación lumínica, resulta factible la identificación de las diferentes texturas que las componen.

La obtención del método de segmentación ha sido posible mediante la rigurosa aplicación del método científico al proceso de investigación. Esta tesis presenta el recorrido trazado a lo largo de la búsqueda del método y los resultados obtenidos con el mismo.

En primer lugar, se sitúa en tema al lector a partir del marco teórico de cada uno de los temas abordados. A continuación tienen lugar las primeras aproximaciones de segmentación logradas mediante el algoritmo y se detallan, así mismo, los indicadores de calidad utilizados para medir cuantitativamente la efectividad del método. Luego son presentadas algunas discusiones que condujeron al perfeccionamiento y refinamiento de la calidad de los resultados. Dicha discusión da lugar a una nueva etapa de experimentación en la que, en particular, se encontró que el par de variables macroscópicas que mejor cuantifica la separación de texturas es entropía-velocidad. Los experimentos numéricos ensayados posibilitaron estimar la calidad de segmentaciones de imágenes de distintas índoles (sintéticas, naturales y mixtas). En promedio, la calidad de las segmentaciones (índice AOM) obtenidas para imágenes mixtas (sintéticas-naturales) es del 95%; mientras que, para imágenes completamente naturales es de 85%. Finalmente, se presenta el modo en que fue realizada la implementación computacional del algoritmo, las conclusiones y los trabajos futuros.

TABLA DE CONTENIDO

AUTÓMATAS DE LATTICE-BOLTZMANN PARA MODELOS DE ILUMINACIÓN DIFUSA APLICADOS A LA DETECCIÓN DE TEXTURAS EN IMÁGENES DIGITALES	1
TABLA DE CONTENIDO	4
ÍNDICE DE FIGURAS	7
1. INTRODUCCIÓN	13
1.1 <i>Presentación del problema</i>	13
1.2 <i>Objetivos</i>	15
1.3 <i>La Tesis</i>	15
2. DETECCIÓN Y SEGMENTACIÓN DE TEXTURAS	17
2.1 <i>Detección y Clasificación</i>	18
2.1.1 Métodos estructurales, sintácticos o geométricos	19
2.1.2 Métodos estadísticos	21
2.1.3 Métodos basados en modelos	25
2.1.4 Métodos espectrales o basados en transformaciones	25
2.2 <i>Segmentación</i>	26
2.2.1 Umbralado.....	27
2.2.2 Métodos orientados a bordes.....	28
2.2.3 Watershed	28
2.2.4 Modelos Deformables	30
3. MODELO DE LATTICE-BOLTZMANN.....	32
3.1 <i>Modelo de Transporte de Luz</i>	33
3.2 <i>Condiciones de contorno</i>	36
3.3 <i>Condiciones iniciales y tolerancia</i>	36
3.4 <i>Detección y segmentación de texturas basada en lattice-boltzmann</i>	36
3.5 <i>Construcción de la Matriz de transición</i>	37
4. VALIDACIÓN DE LOS PRIMEROS EXPERIMENTOS NUMÉRICOS	40
4.1 <i>Variables macroscópicas</i>	40
4.1.1 Validación experimental.....	40

<i>4.2 Distribución Promediada</i>	42
<i>4.2.1 Validación experimental.....</i>	43
<i>4.3 Separación por ángulo.....</i>	46
<i>4.3.1 Validación experimental.....</i>	48
<i>4.4 Separación por Ángulo y Difusión.....</i>	51
<i>4.4.1 Validación experimental.....</i>	52
<i>4.5 Campo de velocidades</i>	55
<i>4.5.1 Validación experimental.....</i>	57
<i>4.6 Espacio físico.....</i>	60
<i>4.6.1 Validación experimental.....</i>	61
<i>4.7 Discusión</i>	66
5. GENERACIÓN DEL MEDIO	68
<i>5.1 Estudio de la problemática en 1D.....</i>	72
<i>5.2 Algoritmo para la detección de texturas en 1D</i>	79
<i>5.3 Caso de estudio: Vecindario promediado con ponderación</i>	81
<i>5.4 Estudio de la problemática en 2D.....</i>	84
<i>5.5 Formalización del Método</i>	84
<i>5.6 Indicadores de calidad de la segmentación.....</i>	90
6. RESULTADOS NUMÉRICOS.....	92
<i>6.1 Caso 1: Textura Homogénea.....</i>	93
<i>6.2 Caso 2: Textura Compuesta con borde Simple</i>	97
<i>6.3 Caso 3: Textura Compuesta con borde Complejo</i>	101
<i>6.4 Caso 4: Texturas Procedurales.....</i>	105
<i>6.4.1 Textura Homogénea</i>	105
<i>6.4.2 Textura Compuesta simple.....</i>	107
<i>6.4.3 Textura Compuesta compleja</i>	111
<i>6.5 Caso 5: Fotografías de Texturas</i>	114
<i>6.6 Caso 6: Imagen con texturas sintético-naturales</i>	117
<i>6.7 Caso 7: imagen con textura natural</i>	121
<i>6.8 Limitaciones y alternativas</i>	125

7.	IMPLEMENTACIÓN COMPUTACIONAL	126
7.1	<i>Proceso de desarrollo</i>	126
7.2	<i>Diseño del sistema</i>	127
7.3	<i>Estructuras de datos principales</i>	130
7.3.1	Látice	130
7.3.2	Matriz del Material.....	131
7.4	<i>Algoritmo de Simulación lattice-Boltzmann</i>	132
8.	CONCLUSIONES Y TRABAJOS FUTUROS	134
	REFERENCIAS	136

ÍNDICE DE FIGURAS

Figura 1 Una textura es un patrón de variaciones locales de intensidad que se repite.....	17
Figura 2 Imágenes con mismo promedio de intensidades, pero con diferente textura.	18
Figura 3 (a) Textura primitiva (b) Patrón generado por la regla $S \rightarrow aS$ (c) patrón de textura bidimensional generado por esta y otras reglas (extraído de González 1996).....	20
Figura 4 Imagen original (izquierda) y binarizaciones a diferentes valores de umbral	27
Figura 5 Resultado de la segmentación utilizando el método de watershed. Fuente Http://biodynamics.ucsd.edu/ir	29
Figura 6 Pasos sucesivos del algoritmo de crecimiento de regiones en imágenes cerebrales. Extraído de http://www.creatis.insa-lyon.fr/~grenier/?cat=8	30
Figura 7 Contorno inicial (amarillo) y el resultado de aplicar un algoritmo de modelos deformables (azul) a la segmentación de una mano.	31
Figura 8 A izquierda, vectores de dirección para un esquema bidimensional de 8 vecinos; a derecha, representación del látice.....	32
Figura 9 A izquierda, imagen base. A derecha, vecindario p de un píxel $[i,j]$	37
Figura 10 Conjunto de prueba 1.....	40
Figura 11 Resultados de Densidad Nij (izquierda) y Entropía Hij (derecha), para el operador Suma aplicado a la imagen 2 del conjunto de prueba	41
Figura 12 Resultados de Densidad Nij (izquierda) y Entropía Hij (derecha), para el operador Resta aplicado a la imagen 3 del conjunto de prueba.	42
Figura 13 Conjunto de prueba. Todos los elementos, a excepción de los extremos, poseen 3 píxeles blancos y 6 negros.	43
Figura 14 Valores promedio de las fi para cada imagen del conjunto de prueba. Operador Suma	44
Figura 15 Valores promedio de las fi para cada imagen del conjunto de prueba. Operador Resta	45
Figura 16 Esquema de separación del espacio angular para los parámetros α y β	47
Figura 17 Textura base del experimento	47
Figura 18 Textura base, mascara y textura final	48

Figura 19 Imagen de prueba	49
Figura 20 Distribución de ángulos para $\alpha = \pi/6$ y $\beta = 0.56$	50
Figura 21 Resultado del método de separación de ángulos con filtro smooth	50
Figura 22 Distribución de ángulos para $\alpha = \pi/6$ y $\beta = 0.56$. Valores de difusión 0%, 100% y 400% (de izquierda a derecha, respectivamente)	53
Figura 23 Esquema de superposición de la segmentación para difusión de 0%, 100% y 400%	54
Figura 24 Valores de AOM para diferentes porcentajes de difusión	55
Figura 25 Textura de prueba para el cálculo del campo de velocidades	56
Figura 26 Gráfica ux,uy para el operador Resta. Imagen de prueba de la Figura 19.....	57
Figura 27 Gráfica ux,uy para el operador Resta. Imagen de prueba de la Figura 19. Difusión=20% (izquierda) y Difusión=60% (derecha)	58
Figura 28 Gráfica ux,uy para el operador Resta. Imagen de prueba de la Figura 25.....	59
Figura 29 Gráfica ux,uy para el operador Suma. Imagen de prueba de la Figura 25.....	60
Figura 30 Textura Homogénea, compuesta simple y compuesta irregular	61
Figura 31 Ángulo y densidad para una textura homogénea	62
Figura 32 Ángulo y densidad para una imagen compuesta simple.....	62
Figura 33 Ángulo y densidad para una imagen compuesta	63
Figura 34 Ángulo y densidad para una imagen compuesta (Operador Suma)	64
Figura 35 Ángulo y entropía para una imagen compuesta (Operador Resta)	65
Figura 36 Ángulo y entropía para una imagen compuesta (Operador Suma)	65
Figura 37 Patrón base (izquierda), imagen de textura homogénea (derecha)	66
Figura 38 Ventana de 3x3 para tres de los vecinos del pixel i,j	67
Figura 39 Ángulo y densidad para una textura homogénea	69
Figura 40 Resultados del algoritmo con vecindario promediado para una textura compuesta	70
Figura 41 Histograma para la textura homogénea (Izquierda) y para la compuesta (Derecha)	71
Figura 42 Vectores de dirección para un esquema unidimensional de 2 vecinos (Izquierda). Representación del látice (Derecha).	72

Figura 43 Matrices de textura para el operador división en 1D	73
Figura 44 Matrices de textura luego de aplicar el promediado por vecindario	74
Figura 45 Matrices de textura luego de la normalización.....	74
Figura 46 Gráfica de la velocidad y la entropía para el caso 1D Homogéneo.....	76
Figura 47 Matrices para la textura homogénea, utilizando la ponderación con el pixel central.....	77
Figura 48 Matrices de características, luego de aplicar el promediado por vecindario	78
Figura 49 Matrices de características normalizadas	78
Figura 50 velocidad y entropía para el caso de estudio (Izquierda) e histograma de velocidades (Derecha)	82
Figura 51 velocidad y entropía (Izquierda) e histograma de velocidades (Derecha) para ruido = 1%.....	83
Figura 52 velocidad y entropía (Izquierda) e histograma de velocidades (Derecha) para ruido = 10%	84
Figura 53 Alternativas de experimentación (Los elementos en cada columna pueden combinarse con cualquiera de las otras columnas)	89
Figura 54 Textura homogénea en escala de grises	94
Figura 55 Ángulo y Entropía (Izquierda), Histograma de Ángulos (derecha) para una textura homogénea	94
Figura 56 Ángulo y Entropía (Izquierda), Histograma de Ángulos (derecha) para una textura homogénea con ruido del 1%.....	95
Figura 57 Ángulo y Entropía (Izquierda), Histograma de Ángulos (Derecha) para una textura homogénea rotada	95
Figura 58 Ángulo y Entropía (Izquierda), Histograma de Ángulos (derecha) para una textura homogénea Rotada con ruido del 1%	96
Figura 59 Imagen compuesta, con dos texturas en escala de gris.....	97
Figura 60 Ángulo y Entropía (Izquierda), Histograma de Ángulos (Derecha) para una textura compuesta	98
Figura 61 Valores de AOM para diferentes ángulos de umbral (inicio en 0 y ventana de 180°) para la imagen sin ruido (izquierda), y con ruido del 1% (derecha).....	98

Figura 62 Ángulo y Entropía (Izquierda), Histograma de Ángulos (Derecha) para una textura compuesta con ruido = 1%	99
Figura 63 Segmentación óptima por umbralado para la imagen compuesta sin ruido (izquierda), y con ruido del 1% (derecha)	100
Figura 64 Segmentación para diferentes ángulos de umbral. El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c), 0.8 (d) y 0.82 (e)	100
Figura 65 Segmentación para diferentes ángulos de umbral (figura con ruido). El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c) y 0.8 (d)	101
Figura 66 Imagen con dos texturas y un borde complejo	102
Figura 67 Histograma de ángulos para la imagen compuesta con borde complejo.....	102
Figura 68 Valores de AOM para diferentes ángulos de umbral para la imagen sin ruido (izquierda), y con ruido del 1% (derecha)	103
Figura 69 Segmentación óptima por umbralado para la imagen compuesta con borde complejo sin ruido (izquierda), y con ruido del 1% (derecha)	104
Figura 70 Segmentación para diferentes ángulos de umbral. El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c), 0.8 (d), 0.89 (e)	104
Figura 71 Segmentación para diferentes ángulos de umbral. Textura con ruido del 1% El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c), 0.8 (d), 0.81 (e)	105
Figura 72 Patrón de base de 7x7 (izquierda) y textura generada a partir de ella (derecha)	106
Figura 73 Gráfica del ángulo y la Entropía, luego de aplicar el algoritmo de detección sobre una textura de 7x7	107
Figura 74 Máscara (izquierda) y Textura compuesta con patrón base de 7x7 (derecha).108	
Figura 75 Gráfica de ángulo y entropía (izquierda) y índice AOM para diferentes ángulos de umbral (derecha) para la segmentación sin presencia de ruido.....	108
Figura 76 Segmentación óptima por umbralado para la imagen compuesta si ruido (izquierda), y con ruido del 1% (derecha)	109
Figura 77 Segmentación para diferentes ángulos de umbral (Imagen sin ruido). El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c), 0.8 (d), 0.9 (e), 0.91 (F).....	110

Figura 78 Segmentación para diferentes ángulos de umbral (ruido 1%). El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c), 0.8 (d), 0.81 (e).....	110
Figura 79 Patrón de tela "pata de gallo" (izquierda) y su equivalente procedural (derecha)	111
Figura 80 Ángulo y entropía para textura "Pata de Gallo" con 1% ruido (izquierda) y con región con 20% de ruido (derecha)	112
Figura 81 Mascara para simular el gastado (izquierda) y Patrón de tela "pata de gallo" con 20% de ruido en zona irregular (derecha).....	112
Figura 82 Índice AOM para la textura “Pata de Gallo” con regiones con ruido.....	113
Figura 83 Segmentación para diferentes ángulos de umbral. El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c), 0.75 (D).....	113
Figura 84 Fotografía de una tela de textura homogénea (Izquierda), ángulo y densidad resultantes de la ejecución del algoritmo (Derecha)	114
Figura 85 Fotografía de una tela de textura compuesta (Izquierda), ángulo y densidad resultantes de la ejecución del algoritmo (Derecha)	115
Figura 86 Índice AOM para diferentes ángulos de umbral para la textura de la Figura 72	116
Figura 87 Segmentación para diferentes ángulos de umbral. El índice AOM es de 0.5 (a), 0.6 (b), 0.7 (c), 0.75 (D).....	116
Figura 88 Imagen de estudio que integra texturas sintéticas y naturales.	117
Figura 89 Entropía vs velocidad para la imagen de la Figura 88, para vecindarios de 9x9 pixeles. Los puntos se encuentran coloreados según la segmentación de referencia.....	118
Figura 90 Entropía vs velocidad para la imagen de la Figura 88, para vecindarios de 11x11 pixeles. Los puntos se encuentran coloreados según la segmentación de referencia.....	119
Figura 91 Selección del tamaño del vecindario y de a, coeficiente principal de la parábola.	120
Figura 92 Segmentación de referencia (Izquierda), segmentación de la Figura 88 correspondiente al a=0.94, AOM máximo para vecindarios de 9x9 pixeles (Derecha). ...	120
Figura 93 - Imagen natural compuesta por textura de pino y eucaliptus (Izquierda), mascara que distingue el pino (negro) del eucaliptus (blanco) (Derecha).	121

Figura 94 - Entropía vs Velocidad para la imagen de la Figura 93 usando un vecindario de 9x9. Los puntos se encuentran coloreados según la segmentación de referencia.	121
Figura 95 Entropía vs Velocidad para la imagen de la Figura 93 usando un vecindario de 17x17. Los puntos se encuentran coloreados según la segmentación de referencia.	122
Figura 96 AOM vs coeficiente principal de la parábola ($A=3.16992$) para distintos vecindarios.	123
Figura 97 Segmentación de la Figura 93 con tamaño de vecindario 9, 11, 13, 15 y 17 pixeles cada lado.	123
Figura 98 Imagen natural compuesta por texturas a distinta resolución (Izquierda). Segmentación lograda por el algoritmo (Derecha).	124
Figura 99 Entropía vs Velocidad para la imagen de la Figura 98. Arriba: vecindario de 17x17 pixeles. Los puntos se encuentran coloreados según la segmentación de referencia.....	124
Figura 100 Diagrama de componentes del sistema	127
Figura 101 Flujo del sistema	128
Figura 102 Representación de la matriz de densidad fi	131
Figura 103 Representación de la matriz de características Mij	132
Figura 104 Pseudocódigo del algoritmo.....	133

1. INTRODUCCIÓN

1.1 PRESENTACIÓN DEL PROBLEMA

La detección de texturas en imágenes digitales facilita tareas difíciles o monótonas de diversa índole, tales como la localización de lesiones cancerosas, fallas en productos manufacturados o materiales naturales defectuosos, zonificación geográfica en base a imágenes satelitales. Una textura generalmente se caracteriza como la repetición de un cierto patrón local en cierta región de una imagen. Para discriminar las texturas de una imagen digital, deben extraerse sus características a través de técnicas específicas que, por lo general, no poseen validez universal y presentan limitaciones inherentes que las hacen inaplicables a la detección de otras nuevas texturas. Por ello, los métodos de detección en imágenes digitales varían ampliamente dependiendo de la aplicación específica, el tipo de la imagen, el ruido, volúmenes parciales y los movimientos en la imagen que también pueden tener consecuencias significativas en el desempeño de los algoritmos de detección.

La detección en imágenes reales requiere de un procesamiento intensivo mediante algoritmos complejos que extraen características locales de las imágenes para identificar, recuperar, clasificar o segmentar las texturas. En este contexto, los autómatas celulares han sido ampliamente usados en el procesamiento de imágenes digitales (Preston *et al.*, 1979; Hernández & Herrmann, 1996; Popovici & Popovici, 2002; Wongthanavasu & Sadananda, 2003; Rosin, 2006). Todas estas aplicaciones se focalizan en imágenes binarias porque la definición de las reglas de los autómatas celulares para el procesamiento en imágenes de 256 tonos de gris es complicada de diseñar. Sus descendientes directos, los autómatas de lattice-Boltzmann no presentan este problema.

Los modelos de lattice-Boltzmann (LBM) son extensamente empleados en el estudio y la simulación de fluidos, por la simplicidad de su implementación y su capacidad para tratar condiciones de contorno complejas (Chen & Doolen, 1998). LBM simula el transporte de partículas, cuya población local evoluciona de acuerdo a sucesivas actualizaciones de estado en una grilla discreta. Desde su origen, el método LBM se ha aplicado a un amplio espectro de dominios, especialmente en la simulación computacional de fluidos. En el área de medicina, por ejemplo, se ha modelado el comportamiento de la sangre para observación del sistema circulatorio humano. Otras aplicaciones, en el contexto de la Computación Gráfica, están relacionadas a la simulación de fenómenos físicos complejos como la animación en tiempo real de fluidos para lograr interactividad.

Recientemente, se han utilizado estas mismas ideas para la creación de modelos de iluminación, adaptando los conceptos de “densidad”, “transporte” y “colisión” utilizados para la simulación de fluidos (Geist *et al.*, 2004). La luz es concebida como un flujo de partículas (fotones) y la cantidad de ellos en una zona del espacio se denomina “intensidad”. La diferencia principal con las partículas de fluidos es que los fotones no interactúan entre sí, sino con el medio en el que se propagan. Además, los fotones tienen comportamientos diferentes, dependiendo de la materia con la que tienen contacto. Los dos fenómenos que rigen su comportamiento se denominan: absorción y dispersión. En la absorción, el fotón es retenido por la materia con la que interactúa en forma de energía interna. En la dispersión, un fotón cambia de dirección al colisionar con los átomos de la materia. Ambos fenómenos están fuertemente relacionados con las características del medio con el que interactúan, dependiendo entonces de las propiedades físicas de estos materiales. Geist *et al.* (2004) encontraron una analogía entre estos procesos y las reglas de lattice-Boltzmann para fluidos.

1.2 OBJETIVOS

En el presente trabajo se elaboró un método para la detección y la segmentación de texturas basado en la aplicación del modelo de lattice-Boltzmann para la propagación de fotones en medios difusos en dos dimensiones. La motivación de esta aproximación tiene en cuenta que la medida de la heterogeneidad óptica de una superficie es relevante para la detección de texturas.

Para investigar la factibilidad de aplicación de los métodos de lattice-Boltzmann a la detección de texturas en imágenes digitales fue necesario desarrollar e implementar un algoritmo que simule las reglas que estos métodos demandan y que, además, se adapte a lo requerido por el procesamiento de imágenes. Por otro lado, también se obtuvo la representación adecuada de estas imágenes, extrayendo correctamente las características principales de una textura. Finalmente, se validó experimentalmente el modelo verificando la calidad de las soluciones.

1.3 LA TESIS

Inicialmente, se elaboró un modelo de lattice-Boltzmann lineal que simula el transporte de fotones. Este modelo necesita de un medio con el cual interactuar (medio por donde se transporta y absorbe la luz), que es construido con la información proveniente de las imágenes cuyas texturas se quiere identificar. En esta etapa se investigaron diferentes técnicas para realizar esta tarea, intentando extraer de las imágenes la mayor información sobre su composición.

A continuación, se elaboraron numerosos experimentos que exploraron diferentes alternativas (modificando parámetros y características del algoritmo) para evaluar el comportamiento numérico del modelo frente a distintos tipos de texturas. Luego, tomando

medidas estadísticas y elaborando diferentes vistas de estos resultados, se verificaron las hipótesis planteadas al comienzo del trabajo y se estudió en detalle la dinámica de los casos unidimensionales (con posible aplicación al estudio de patrones repetitivos en señales). Finalmente, se obtuvo un método por el cual se demostró la factibilidad de la detección y segmentación en un amplio rango de texturas.

2. DETECCIÓN Y SEGMENTACIÓN DE TEXTURAS

Las texturas son patrones homogéneos a la vista que se perciben tanto en la naturaleza como en entornos virtuales. Están constituidas por pequeños patrones que se repiten y producen la sensación de uniformidad (Figura 1). En un lenguaje más formal, una textura es una región macroscópica estructurada de una imagen, que presenta propiedades locales constantes, lentamente variables o aproximadamente periódicas (Hájek *et al.*, 2006).

De la definición anterior se desprende que, las características de una textura están dadas por la distribución espacial de las intensidades y no por el promedio de ellas. La Figura 2 muestra cómo pueden coexistir diferentes texturas con un mismo promedio de intensidades. Esta distinción es importante en la tarea de detección de texturas, ya que es preciso contar con algoritmos que sean parcialmente independientes de las intensidades que componen la imagen.

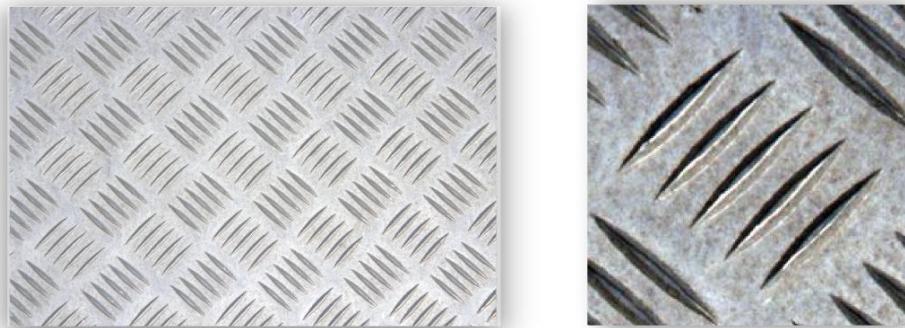


FIGURA 1 UNA TEXTURA ES UN PATRÓN DE VARIACIONES LOCALES DE INTENSIDAD QUE SE REPITE.

Las texturas son definidas a partir de parámetros matemáticos que se computan basándose en la distribución de gris de píxeles o vértices, y que acaban por definir la estruc-

tura subyacente de los objetos que componen la imagen. En la presente sección se enumeran las principales estrategias existentes para la caracterización de texturas.

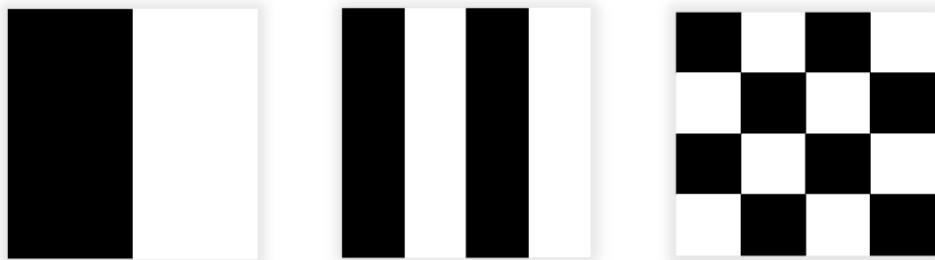


FIGURA 2 IMÁGENES CON MISMO PROMEDIO DE INTENSIDADES, PERO CON DIFERENTE TEXTURA.

En años recientes se han propuesto varias técnicas que son utilizadas para medir diferentes propiedades de textura. Algunos estudios se abocan a la cuantificación de los niveles de gris de una imagen, mientras que otros estudian la interrelación entre los píxeles y las propiedades espectrales de la misma. El análisis de texturas se divide, principalmente, en dos áreas importantes que estudiaremos a continuación:

- detección y clasificación;
- segmentación.

2.1 DETECCIÓN Y CLASIFICACIÓN

La detección y clasificación se ocupa de identificar una región texturada de una imagen, de entre un conjunto de texturas diferentes. Cada una de estas regiones posee características únicas que las hacen distinguibles del resto del conjunto.

Existen diferentes formas de realizar esta tarea que se pueden clasificar en cuatro grandes grupos:

- *Estructural*: una textura es concebida como un conjunto de *texels* (unidad espacial fundamental de la que se compone un textura [Glassner, 1989]). Estos *texels* están relacionados en forma repetida o regular en la imagen.
- *Estadística*: una textura se considera una medida cuantitativa de la distribución de intensidades de una región. Se suelen utilizar vectores que recogen las características particulares de cada textura, denominados *vectores característicos*.
- *Modelado*: las texturas se asocian a modelos que las identifican, construidos especialmente para ellas.
- *Métodos espectrales o basados en transformaciones*: las técnicas espectrales están basadas en propiedades del espectro de Fourier u otros, y son usadas primariamente para detectar la periodicidad global presente en la imagen identificando concentración grande de la energía y picos estrechos en el espectro (Rodríguez, 2008).

Los métodos estadísticos suelen ser más efectivos cuando se trabaja con regiones pequeñas. En cambio, si el tamaño de la textura primitiva es grande, suele ser más efectiva la utilización del análisis estructural o modelado, que identifican la forma y las reglas que gobiernan la posición de las primitivas.

A continuación se detallan brevemente algunos de los métodos que se utilizan para la detección y clasificación de texturas.

2.1.1 MÉTODOS ESTRUCTURALES, SINTÁCTICOS O GEOMÉTRICOS

Los métodos estructurales, sintácticos o geométricos representan la morfología de una textura a través de primitivas bien definidas (generalmente denominadas *micro textu-*

ras), que intentan describir las primitivas y las reglas que gobiernan la organización espacial dentro de la misma (*macro textura*).

En primer lugar, para definir una textura por un método sintáctico, se deben establecer las primitivas y reglas de localización. La elección de la primitiva (de entre un conjunto) y la probabilidad de que la primitiva elegida sea ubicada en un punto en particular puede ser una función del punto o de las primitivas cercanas al mismo.

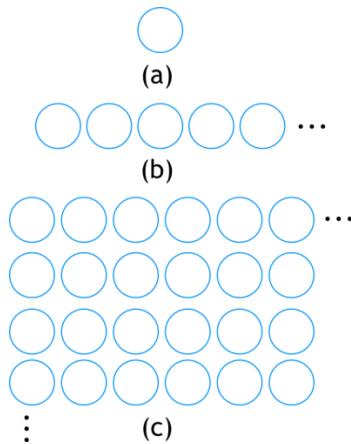


FIGURA 3 (A) TEXTURA PRIMITIVA (B) PATRÓN GENERADO POR LA REGLA $S \rightarrow aS$ (C) PATRÓN DE TEXTURA BIDIMENSIONAL GENERADO POR ESTA Y OTRAS REGLAS (EXTRAÍDO DE GONZÁLEZ 1996).

La Figura 3 corresponde a un ejemplo extraído de González & Woods (1996) y presenta la primitiva (la circunferencia) y la regla de construcción definida como:

$$S \rightarrow aS$$

que indica que, el símbolo S se vuelve a escribir como aS (por ejemplo, tres aplicaciones de esta regla producirían la cadena $aaaS$). Si a es considerada como un círculo, la regla permite generar un patrón de texturas como la que muestra la Figura 3 (b). Finalmente, si se anexan más reglas pueden construirse texturas como la de la Figura 3 (c).

Una de las principales ventajas de estos métodos es que proveen una buena descripción en términos simbólicos de la imagen. Sin embargo, tal como pudo ser apreciado en el ejemplo anterior, resultan mejores para su síntesis que para su análisis, dado el carácter abstracto de las descripciones.

Los métodos estructurales más utilizados están basados en el uso de morfología matemática. La misma demuestra, por ejemplo, especial utilidad en el análisis de imágenes óseas, para la detección de cambios en la microestructura de los huesos (Hájek *et al.*, 2006).

2.1.2 MÉTODOS ESTADÍSTICOS

La distribución espacial de valores de gris es una cualidad que define la textura, por ello, analizándola es posible computar características locales de la textura. Las propiedades estadísticas de una textura son aquellas que describen regiones estadísticamente homogéneas de esa textura.

Estas técnicas no intentan interpretar de manera explícita la estructura jerárquica de la textura, sino que las representan a través de propiedades que definen la distribución y la relación de los valores de niveles de gris en la imagen. Por lo general, alcanzan mayores índices de discriminación que los métodos estructurales o basados en transformaciones, y son extensamente utilizados en aplicaciones médicas (Castellano *et al.*, 2004).

Generalmente, estas técnicas construyen un vector de *densidad de probabilidad* $P_i(z)$ que indica cuánto se repite cada nivel de gris en la imagen (i). Es decir que, si la imagen contiene k -niveles de grises, $P_i(z)$ será un vector de k -elementos, uno por cada

nivel de gris. Las medidas estadísticas que se pueden extraer de este vector, ofrecen información sobre la textura en estudio.

Las características de la textura se cuantifican a partir de estadísticas de primer o segundo orden.

ESTADÍSTICAS DE PRIMER ORDEN

Son las estadísticas derivadas del vector de densidad. Representan la probabilidad de obtener un determinado nivel de gris dada una ubicación aleatoria de la imagen.

La varianza y la desviación estándar son algunas de las estadísticas que conforman este conjunto:

1. Varianza definida como :

$$\sigma_i^2 = \frac{1}{N} [(z - \mu^2) \cdot P_i(z)]$$

donde μ es la intensidad de gris media de la imagen, y z la variable aleatoria con dominio en el vector de densidad de tamaño N .

2. Desviación estándar del nivel de gris definida como:

$$\sigma_i = \sqrt{\frac{1}{N} [(z - \mu^2) \cdot P_i(z)]}$$

La ventaja de estas técnicas radica en que es posible adquirir propiedades de las texturas, con un conocimiento mínimo de las mismas, así como también con un escaso procesamiento de la imagen.

Como desventaja existe el hecho de que estos parámetros sólo dependen de cada valor de los píxeles/vóxeles, y no de la interacción o co-ocurrencia con valores de los píxe-

les/vóxeles vecinos, con lo cual son los menos utilizados para caracterizar la textura de una región de interés. En (Hájek *et al.*, 2006) se hace referencia al empleo de estadísticas correspondientes a histogramas locales para determinar qué píxeles pertenecen a un área que está siendo segmentada y cuáles no.

ESTADÍSTICAS DE SEGUNDO ORDEN

Estudian las estadísticas de las texturas, pero teniendo en cuenta la distribución espacial de las intensidades. Para ello, se calculan a partir de las matrices de co-ocurrencia de niveles de gris (*Gray-Level Co-occurrence Matrix o GLCM*) y las matrices de corridas de niveles de gris (*Run-length Matrix, RLM*).

- GLCM

La GLCM o matriz de co-ocurrencia, es una matriz simétrica que estima la probabilidad de que dos píxeles ubicados a una distancia $\Delta x, \Delta y$ en la imagen, tengan el mismo valor, y se construye considerando la relación entre pares de píxeles para cada dirección, y tabulando la frecuencia de varias combinaciones de niveles de gris en la región de estudio.

Considérense un par de píxeles ubicados a $\Delta x, \Delta y$ el uno del otro. El elemento $C_{\Delta x, \Delta y}(i, j)$ de la matriz de co-ocurrencia determina la cantidad de estos pares cuyos píxeles toman los valores de gris i y j . La formulación matemática viene dada por la ecuación:

$$C_{\Delta x, \Delta y}(i, j) = \sum_{p=1}^n \sum_{q=1}^m \begin{cases} 1 & \text{si } I(p, q) = i, y \quad I(p + \Delta x, q + \Delta y) = j \\ 0 & \text{en cualquier otro caso} \end{cases}$$

La matriz construida se trata de una matriz de dimensión $N \times N$, siendo N el número de niveles de grises de la imagen original. Los elementos cercanos a la diagonal principal

corresponden a niveles de grises similares, mientras que los que están más alejados de la diagonal, son niveles de grises diferentes. Si se divide cada elemento por la cantidad total de ocurrencias, se obtiene una nueva matriz, denominada de probabilidad conjunta.

Sobre estas matrices es posible calcular valores de energía, contraste, correlación, entropía, la inversa de la diferencia de momentos y el momento diagonal, valores que se utilizan como caracterizadores de la textura, entre otros.

Si bien estos índices resultan efectivos, el cómputo de esta matriz es extremadamente costoso (Hájek *et al.*, 2006), algo que complica la aplicación de esta técnica en un proceso de segmentación basado en crecimiento de regiones, por ejemplo, donde la matriz debería calcularse dentro de una ventana de análisis toda vez que intenta incorporarse un nuevo píxel a la región segmentada.

- RLM

La RLM puede computarse para evaluar la “aspereza” de una textura en una dirección predeterminada. Una corrida de nivel de gris consiste en un conjunto de píxeles consecutivos y colineares en una dirección dada “ θ ”. La RLM es construida de manera tal que $P_\theta(i,j)$ es el número de corridas de un nivel de gris i de j píxeles de largo. De acuerdo a Kassner & Thornill (2010), la RLM no ha dado muestras de ser más eficiente que la GLCM sino en ciertos casos aislados. Sin embargo, en algunos trabajos como Padma & Sukanesh (2011), se destaca su utilidad para las aplicaciones de segmentación de tumores en tomografías cerebrales, por sobre el uso de estadísticas basadas en GLCM.

2.1.3 MÉTODOS BASADOS EN MODELOS

Estos métodos se basan en la construcción de un modelo, cuyos parámetros estimados sobre regiones más pequeñas de la imagen, describen las cualidades de la textura (Epifanio, 2002). Para ello, utilizan dimensión fractal o descripciones estocásticos como los campos aleatorios de Markov, Gibbs o Wold para interpretar la textura de una imagen, basándose, respectivamente, en un modelo generativo de la imagen o en un modelo estocástico. Los parámetros del modelo se estiman y luego son utilizados para el análisis de la imagen. Una desventaja de estos modelos, es que resultan muy costosos en cuanto al tiempo computacional que requiere su cálculo.

En particular, la utilización de enfoques basados en fractales ha demostrado ser de gran utilidad para modelar algunas texturas naturales (Hájek *et al.*, 2006), así como también para el análisis y discriminación de texturas. Sin embargo, y dado que los resultados carecen de una orientación inicial, no resultan útiles para la descripción de estructuras locales.

2.1.4 MÉTODOS ESPECTRALES O BASADOS EN TRANSFORMACIONES

Los patrones de textura de una imagen varían su apariencia dependiendo de la escala de la imagen. Los métodos espectrales o basados en transformaciones representan a la imagen en un espacio cuyo sistema de coordenadas tiene una interpretación que está estrechamente relacionada con las características de una textura (por ejemplo, la frecuencia o el tamaño) (Hájek *et al.*, 2006). Algunos de los métodos están basados en la transformada de Fourier, de Wavelet y los filtros de Gabor.

Los métodos basados en la transformada de Fourier suelen no ser muy efectivos, dado que carecen de localización espacial. Los basados en filtros de Gabor tienen mayor

precisión espacial. Su uso está limitado en la práctica debido a que normalmente no existe una única resolución para el filtro con la cual pueda localizarse una estructura espacial en texturas naturales. Además, su no-ortogonalidad hace que las características obtenidas a menudo sean redundantes a diferentes escalas o canales (Hájek *et al.*, 2006). No obstante, los filtros de Gabor se utilizan frecuentemente para la segmentación de texturas.

La transformada Wavelet, otra de las técnicas usuales para la segmentación, tiene grandes ventajas comparada con la transformada Gabor. En efecto, la variación de la resolución espacial permite la representación de la textura en la escala más conveniente y, además, existe un amplio rango de funciones wavelets que pueden elegirse, de manera que puede escogerse la que mejor se adapte al problema que está queriendo resolver.

2.2 SEGMENTACIÓN

La segunda área importante dentro del análisis de texturas es la segmentación. La tarea básica de la segmentación de texturas es obtener una separación espacial de las partes de una imagen que contiene regiones de textura similar. Algunas de sus aplicaciones se ocupan del reconocimiento de caras, iris y huellas digitales (biometría); en medicina: la localización de lesiones cancerosas, mediciones óseas y de tejido en general como también el diagnóstico de enfermedades; en visualización computacional y en los sistemas de control de tráfico.

En el presente trabajo se propone una nueva técnica para la segmentación de imágenes, su desarrollo incorpora al algoritmo de segmentación por umbralado en la etapa de separación de las texturas que componen a la imagen digital. Se pretende que el método propuesto para la detección proporcione nuevas imágenes, de segmentación sencilla por algunos de los métodos más conocidos en el área.

A continuación, se incluye un compendio de los algoritmos de segmentación más difundidos en el área de procesamiento de imágenes, junto a una descripción general de cada uno de ellos.

2.2.1 UMBRALADO

La técnica de umbralado es una de las técnicas de segmentación más simples. Es utilizada para transformar cualquier imagen a binaria. De este modo, los objetos de interés pueden ser bien diferenciados del fondo, que posee otro valor de intensidad uniforme pero diferente al del objeto.

El proceso consiste en comparar cada píxel de la imagen con un valor umbral (*threshold*); si la intensidad del pixel es mayor, entonces el mismo es considerado parte de la región de interés; en caso contrario, formará parte del fondo, y será descartado.

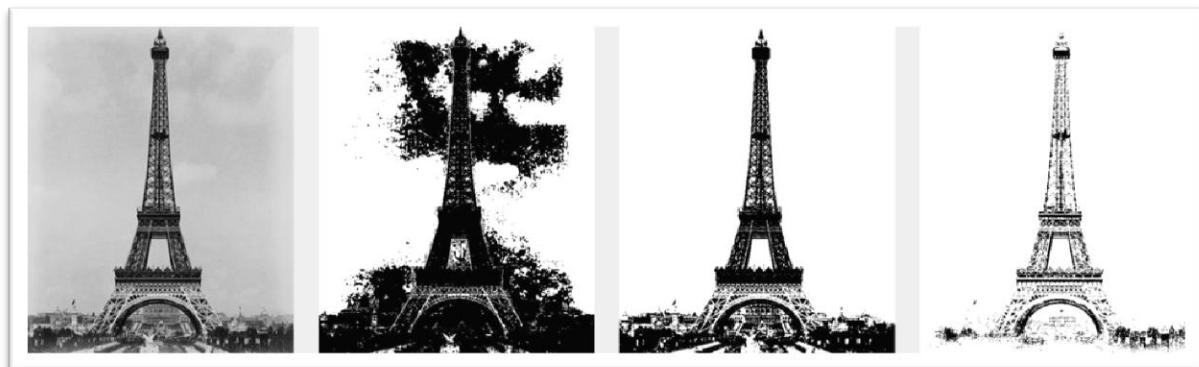


FIGURA 4 IMAGEN ORIGINAL (IZQUIERDA) Y BINARIZACIONES A DIFERENTES VALORES DE UMBRAL

La Figura 4 ilustra el efecto del umbralado de una imagen en escala de grises. A primera vista puede observarse la binarización de las imágenes resultantes. El balance de blancos y negros dependerá tanto de la imagen original, como del umbral elegido.

Uno de los problemas de este método, es su alta dependencia respecto del valor del umbral. El mismo puede ser especificado manualmente por el usuario del algoritmo, o

calcularse mediante algún método automático (por ejemplo, analizando el histograma de la imagen y determinando los valles que se presentan en el mismo).

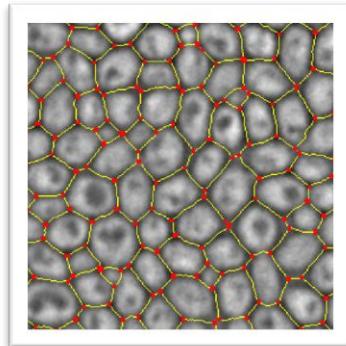
2.2.2 MÉTODOS ORIENTADOS A BORDES

Esta clase de métodos se basa en la búsqueda de diferencias entre píxeles. En efecto, es posible inferir la presencia de bordes cuando se detectan saltos abruptos en algún parámetro de la imagen como, por ejemplo, la intensidad de los píxeles.

Una de las técnicas más comunes consiste en utilizar un operador gradiente (por ejemplo, un filtro de Sobel [Engel, 2006]) para realizar la presencia de los bordes y, luego, aplicar el umbralado sobre la imagen resultante. En imágenes ruidosas los resultados no suelen ser buenos, por lo que es recomendable suavizar las imágenes antes de someterlas a este proceso.

2.2.3 WATERSHED

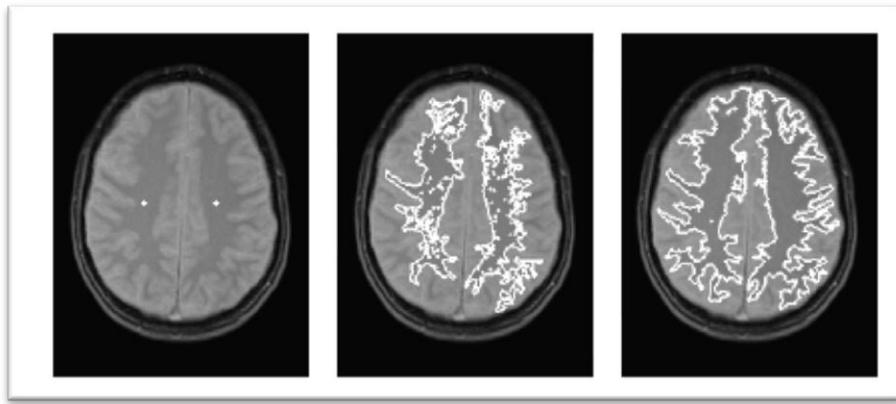
El método de segmentación *watershed*, conocido como “línea de división de aguas”, supone que la imagen en escalas de grises que está siendo analizada constituye un mapa topográfico en el cual las intensidades de los píxeles representan diferentes alturas. Dicho mapa es “inundado” desde sus puntos mínimos, simulando la creación de lagunas artificiales en los distintos valles. Las curvas donde dichas lagunas comienzan a unirse son consideradas como *watersheds*, y se interpretan como los límites de las diversas regiones (ver Figura 5).



**FIGURA 5 RESULTADO DE LA SEGMENTACIÓN UTILIZANDO EL MÉTODO DE WATERSHED. FUENTE
[HTTP://BIODYNAMICS.UCSD.EDU/IR](http://biodynamics.ucsd.edu/ir)**

Los métodos basados en regiones intentan conectar píxeles vecinos basándose en las similitudes entre ellos. El objetivo es construir regiones conectadas que correspondan a los objetos buscados.

Uno de los métodos más conocidos basado en regiones es *region growing*. Este método es inicializado con uno o varios píxeles semilla por región, a partir de los cuales comienzan a agregarse progresivamente vecinos que cumplan con un criterio de aceptación determinado. El criterio de aceptación dependerá del dominio de estudio, pero generalmente es planteado en base a características tales como brillo, color, intensidad, textura, gradiente, propiedades geométricas, etc. El proceso concluye cuando no existen vecinos por agregar. Este criterio puede ser también, el resultado de un detector de texturas que guía al algoritmo de crecimiento de regiones.



**FIGURA 6 PASOS SUCESIVOS DEL ALGORITMO DE CRECIMIENTO DE REGIONES EN IMÁGENES CEREBRALES. EXTRAÍDO DE
[HTTP://WWW.CREATIS.INSALYON.FR/~GRENIER/?CAT=8](http://www.creatis.insa-lyon.fr/~grenier/?cat=8)**

En la Figura 6 puede observarse, para sucesivos pasos del algoritmo, la segmentación de una imagen de un corte cerebral. Las semillas están ubicadas en cada lóbulo y en cada imagen, se aprecia cómo estas se expanden sobre áreas de similares características.

En este tipo de algoritmos, la elección de los píxeles semilla o de los parámetros del modelo puede afectar a la segmentación y dar soluciones diferentes. La variedad de imágenes en que esta técnica puede resultar útil es muy amplia y los resultados en imágenes problemáticas, como aquellas que presentan ruido y bordes difusos, suelen ser mejores que los obtenidos con otros métodos (como los orientados a borde).

2.2.4 MODELOS DEFORMABLES

La segmentación por Modelos Deformables (también conocido como “contornos activos” o “snakes”) es un método de segmentación basado en un contorno continuo que se adapta de manera elástica a la estructura de la región de interés. El control de contorno es llevado a cabo a través de una función de energía de manera análoga a lo que sucede en un sistema físico de interacción de fuerzas.

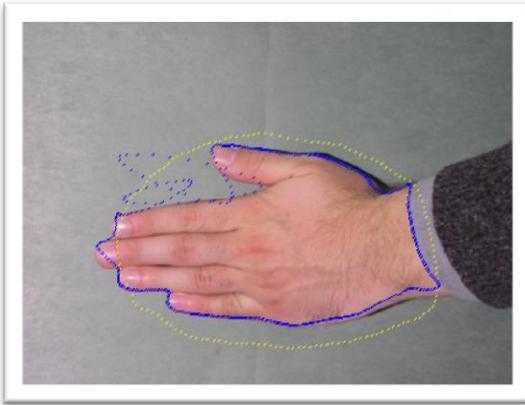


FIGURA 7 CONTORNO INICIAL (AMARILLO) Y EL RESULTADO DE APLICAR UN ALGORITMO DE MODELOS DEFORMABLES (AZUL) A LA SEGMENTACIÓN DE UNA MANO.

La posición inicial del contorno es especificada por un conjunto de puntos de control iniciales, con un espaciado aproximadamente uniforme, que deben colocarse lo más cerca posible de la región de interés. La posición de cada uno de los puntos de control es actualizada en las sucesivas iteraciones del algoritmo, dependiendo del valor devuelto por la función de energía. La Figura 7 ejemplifica con la aplicación de este algoritmo a la segmentación de una mano. En amarillo aparece el contorno inicial (delimitado manualmente) y en azul puede observarse el resultado final ofrecido por el modelo deformable.

Una de las ventajas principales de este método, es que el resultado obtenido está dado por una curva cerrada y suave, a diferencia de otros métodos, como region growing, que devuelven fronteras en forma de escalón. Sin embargo, si el contorno a ser detectado presenta bordes con muchas concavidades, los resultados obtenidos no serán los mejores.

3. MODELO DE LATTICE-BOLTZMANN

Los métodos lattice-Boltzmann (LBM) surgieron inicialmente como una curiosidad teórica de la física estadística, cuando se descubrió que los promedios estadísticos de los autómatas celulares operando en grillas regulares tienden a la solución de las ecuaciones de Navier-Stokes (Constantin P, 1988) de fluidos viscosos en el límite continuo (Currie, 1974). Esencialmente, LBMea una forma simplificada de la ecuación de transporte de Boltzmann, en la que se discretizan todas las variables independientes: espacio, tiempo y velocidad (Chopard *et al.*, 2002). El espacio es dividido en una grilla regular de celdas, el tiempo en una sucesión de pasos finitos y la velocidad en un conjunto finito de vectores (ver Figura 8). Las partículas pueblan dichas grillas y la concentración de estas puede variar en cada paso discreto de tiempo. Las partículas se intercambian entre celdas vecinas de acuerdo a su velocidad.

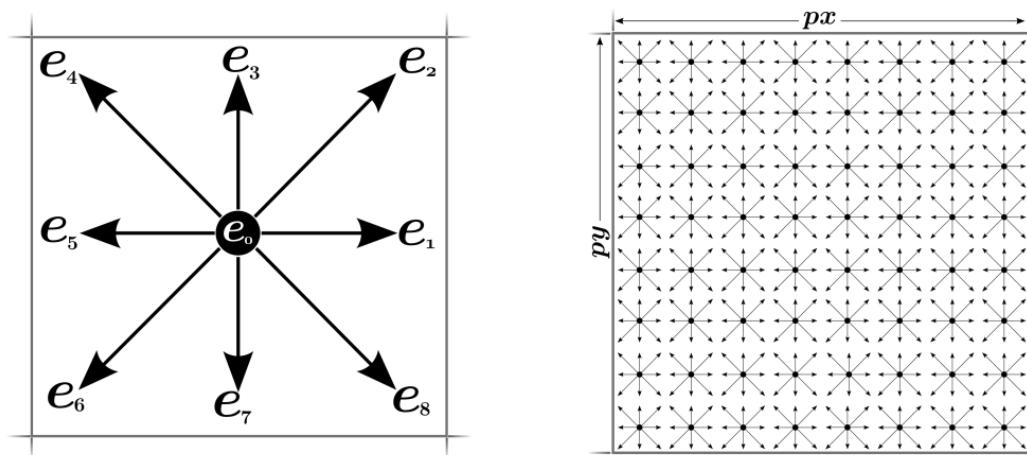


FIGURA 8 A IZQUIERDA, VECTORES DE DIRECCIÓN PARA UN ESQUEMA BIDIMENSIONAL DE 8 VECINOS; A DERECHA, REPRESENTACIÓN DEL LÁTICE

Los modelos de lattice-Boltzmann están regidos por dos reglas fundamentales: colisión y transporte. El transporte (o propagación) define cómo se moverán las partículas entre celdas adyacentes en la dirección de sus velocidades. La colisión hace que las partículas cambien localmente su dirección. En simulación de fluidos se impone, además, que la regla de colisión conserve características físicas como la masa y la cantidad de movimiento. La variable de estado de una red de Boltzmann es la función de distribución f , que representa el número de partículas contenidas en una celda x en el tiempo t moviéndose con velocidad \vec{e}_i . La evolución de f está dada por la ecuación:

$$f_i(\vec{x} + \vec{e}_i \Delta x, t + \Delta t) = \Omega_i^t[f(\vec{x}, t)], i = 0, 1, \dots, l$$

ECUACIÓN 1 EVOLUCIÓN DE LA DISTRIBUCIÓN F

donde $\Omega_i[f(\vec{x}, t)]$ es el operador de colisión y l es cantidad de vecinos de la celda i . En fluidos $\Omega_i[f(\vec{x}, t)]$ se calcula como:

$$\Omega_i^t(\vec{x}, t + \Delta t) = f_i(\vec{x}, t) + \frac{1}{\tau} (f_i^{eq} - f_i)$$

ECUACIÓN 2 OPERADOR DE COLISIÓN

Siendo f_i^{eq} la función de equilibrio (que es función de promedios locales de f_i) y τ un factor de relajación relacionado con la viscosidad del fluido.

3.1 MODELO DE TRANSPORTE DE LUZ

Recientemente Geist *et al.* (2004) propusieron una técnica para simulación de procesos de iluminación basada en LBM, adaptando los conceptos de “densidad”, “transporte” y “colisión” utilizados para la simulación de fluidos. La luz es concebida como un flujo de

partículas (fotones) y la cantidad de ellos en una zona del espacio se denomina “intensidad”. La diferencia principal con fluidos es que los fotones no interactúan entre sí, sino con el medio en el que se propagan.

Tal como es propuesto en (Geist *et al.*, 2004), se distinguen dos procesos principales que determinan la interacción de fotones con la materia: absorción y dispersión.

El proceso de dispersión lumínica en un determinado material, viene dado por la ecuación general de transporte de Boltzmann que describe el estado estacionario del fenómeno:

$$(\vec{\omega} \cdot \nabla + \mu_t) f(\vec{x}, \vec{\omega}) = \mu_s \int_{4\pi} M(\vec{\omega}', \vec{\omega}) f(\vec{x}, \vec{\omega}') d\vec{\omega}'$$

ECUACIÓN 3 ECUACIÓN GENERAL DE BOLTZMANN QUE DESCRIBE EL ESTADO ESTACIONARIO DEL SISTEMA

donde $f(\vec{x}, \vec{\omega})$ es el flujo de fotones en la posición \vec{x} propagándose en dirección $\vec{\omega}$ y la probabilidad de que un fotón entrante en \vec{x} con dirección $\vec{\omega}'$ salga de \vec{x} con dirección $\vec{\omega}$ está dada por $M(\vec{\omega}', \vec{\omega})$. La Ecuación 3 puede discretizarse sobre una grilla espacial regular, con actualización discreta en pasos de tiempo, y con un conjunto finito de direcciones.

En este caso, la ecuación de actualización es la siguiente:

$$f_i(\vec{x} + \vec{e}_l, t + 1) = \sum_{j=0}^l M_{ij} f_j(\vec{x}, t)$$

ECUACIÓN 4 ECUACIÓN DE ACTUALIZACIÓN DISCRETA DEL SISTEMA

Donde $f_i(\vec{x}, t)$ es la población de fotones en \vec{x} en el paso t que se desplazan en la dirección \vec{e}_l , l es la cantidad de vecinos de cada celda y M es la matriz estocástica de probabilidad.

dad de transición entre direcciones que en general es dependiente del material del medio y de los parámetros de la grilla.

Una vez construida la matriz, es iterada la ecuación de Boltzmann (Geist *et al.*, 2004) mediante simulación numérica, hasta que la variación de la distribución $f_i(\vec{x}, t)$ es cercana a cero (convergencia). A partir de la distribución f_i se construyen nuevas vistas de los datos, variables macroscópicas, que muestran diferentes perspectivas de los mismos. En efecto, se estudiaron los indicadores de densidad Ecuación 5 (a), velocidad Ecuación 5 (b), módulo de la velocidad Ecuación 5 (c), entropía Ecuación 5 (d), ángulo Ecuación 5 (e) y energía interna Ecuación 5 (f).

$$\text{Densidad: } N_{ij} = \sum_{k=0}^l f_{ijk} \quad (\text{a})$$

$$\text{Velocidad: } \overrightarrow{u_{ij}} = \sum_{k=0}^l f_{ijk} \overrightarrow{e_k} \quad (\text{b})$$

$$\text{Módulo de la Velocidad: } u_{ij} = |\overrightarrow{u_{ij}}| \quad (\text{c})$$

$$\text{Entropía: } H_{ij} = - \sum_{k=0}^l p_{ijk} \ln p_{ijk}, \quad p_{ijk} = \frac{f_{ijk}}{N_{ij}} \quad (\text{d})$$

$$\text{Ángulo: } \theta_{ij} = \text{atan} \left(\frac{uy_{ij}}{ux_{ij}} \right), 0 \text{ si } uy_{ij} = 0 \quad (\text{e})$$

$$\text{Energía interna: } U_{ij} = \sum_{k=0}^l f_{ijk} \epsilon_{ijk}, \quad \epsilon_{ijk} = e_{ijk}^2 - u_{ij}^2 \quad (\text{f})$$

ECUACIÓN 5 VARIABLES MACROSCÓPICAS

3.2 CONDICIONES DE CONTORNO

Para poder aplicar la Ecuación 4 a una imagen, aún resta definir las condiciones de contorno, ya que celdas situadas en el contorno de la grilla no tienen celdas vecinas de donde obtener las distribuciones en el paso de transporte.

Para este estudio, la imagen de base se consideró con bordes periódicos, es decir, los píxeles situados en bordes opuestos son vecinos entre sí (forma toroidal). Esta elección no es casual, y responde al hecho de que nos encontraremos trabajando con texturas simples y aisladas, de tamaño pequeño. Utilizando estas condiciones de contorno, los resultados de los experimentos también serán válidos para diferentes escalas de las imágenes de prueba.

3.3 CONDICIONES INICIALES Y TOLERANCIA

La simulación requiere de una inicialización de la distribución de partículas f . Este valor es elegido arbitrariamente, y es un parámetro más del algoritmo. Además, debe ser establecido el valor de error tolerado, para poder determinar la convergencia de la simulación.

3.4 DETECCIÓN Y SEGMENTACIÓN DE TEXTURAS BASADA EN LATTICE-BOLTZMANN

En el método de detección propuesto es clave la formulación de la matriz de transición M ; ya que, es la encargada de transmitir la información contenida en la imagen digital al proceso de simulación de fotones. La definición de M es arbitraria; por tanto, se estudiaron diferentes alternativas y se realizaron pruebas exhaustivas con todas ellas, hasta conseguir una configuración de M que resultara apropiada en la mayoría de los casos.

3.5 CONSTRUCCIÓN DE LA MATRIZ DE TRANSICIÓN

El proceso de detección se inicia ideando la matriz de transición M con información extraída de la imagen digital estudiada. Así, para cada píxel (i, j) se crea una entrada en $M[i, j]$ de dimensiones $[l \times l]$ (donde l es la cantidad de vecinos de cada celda de la grilla, incluyendo el pixel central), que contiene las probabilidades de cambio de dirección de la luz. Para definir esas probabilidades se exploraron operaciones aritméticas arbitrarias realizadas entre los valores de intensidad en la imagen del vecindario p del pixel (i, j) . Estas operaciones simples, en principio, permitirían dejar al descubierto las características de la textura analizada. Las operaciones exploradas en este trabajo son suma, producto, módulo-resta y división.

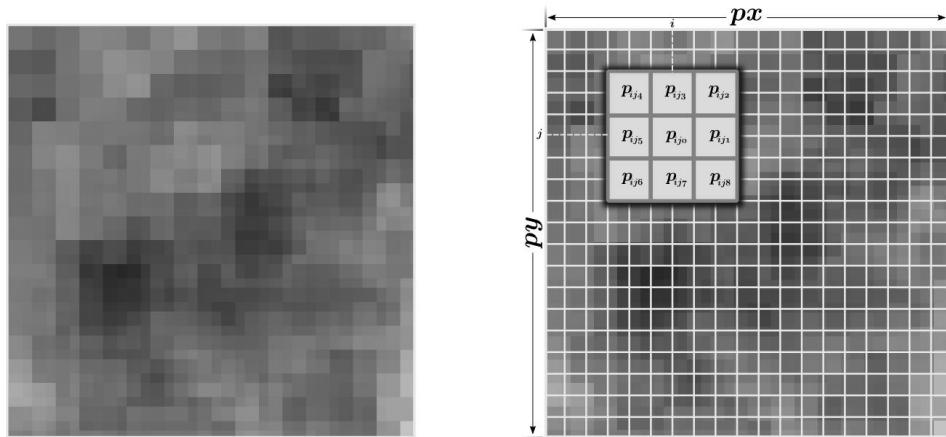


FIGURA 9 A IZQUIERDA, IMAGEN BASE. A DERECHA, VECINDARIO p DE UN PÍXEL $[i, j]$

De esta forma, cada celda de la matriz es constituida según la Ecuación 6, Ecuación 7 y Ecuación 8.

$$\text{Suma: } M[i, j, k, n] = p_{ijk} + p_{ijn}$$

ECUACIÓN 6 REGLA SUMA PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Producto: } M[i, j, k, n] = p_{ijk} \cdot p_{ijn}$$

ECUACIÓN 7 REGLA PRODUCTO PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Módulo - Resta: } M[i, j, k, n] = |p_{ijk} - p_{ijn}|$$

ECUACIÓN 8 REGLA MÓDULO-RESTA PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

donde p_{ijk}, p_{ijn} son intensidades de píxeles vecinos a i, j y $0 \leq n, k \leq l$ (ver Figura 9).

Sin embargo, las ecuaciones precedentes no están completas. $M[i, j]$ debe ser además, una matriz estocástica, razón por la cual, es preciso normalizar las columnas de la siguiente manera:

$$\text{Suma: } M[i, j, k, n] = \frac{p_{ijk} + p_{ijn}}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 9 REGLA SUMA NORMALIZADA PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Producto: } M[i, j, k, n] = \frac{p_{ijk} * p_{ijn}}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 10 REGLA PRODUCTO NORMALIZADO PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Módulo - Resta: } M[i, j, k, n] = \frac{|p_{ijk} - p_{ijn}|}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 11 REGLA MÓDULO-RESTA NORMALIZADA PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

donde p_{ijk}, p_{ijn} son intensidades de píxeles vecinos a i, j y $0 \leq n, k \leq l$.

4. VALIDACIÓN DE LOS PRIMEROS EXPERIMENTOS NUMÉRICOS

4.1 VARIABLES MACROSCÓPICAS

El primero de los experimentos realizados fue pensado para estudiar el comportamiento del algoritmo frente a casos simples elementales cuyo seguimiento puede ser rea- lizado manualmente (sin necesidad del cálculo computacional).



FIGURA 10 CONJUNTO DE PRUEBA 1

La simulación se realizó sobre el conjunto de imágenes 1 (ver Figura 10) y se obtuvie- ron tanto los datos de la distribución f_i , como también los valores de las variables macro- cóbicas.

El conjunto estudiado está compuesto por tres imágenes cuadradas, con sólo dos in- tensidades diferentes de gris (binarias), y una completamente blanca que sirve de refe- rencia. El objetivo de esta primera etapa fue observar cómo se comportaba el algoritmo frente a la presencia de bordes y cuáles eran las diferencias obtenidas al procesar la dis- tribución f_i para conseguir las variables macroscópicas (entropía, velocidad, etc.).

4.1.1 VALIDACIÓN EXPERIMENTAL

Se esperaba que, tanto la distribución f_i como las variables calculadas a partir de ella (para cualquier operador de construcción de M), tuvieran valores constantes. Efectiva-

mente, esto fue corroborado en los experimentos, y la utilidad de este resultado radicó en que pudo ser utilizado como punto de referencia para el análisis del resto del conjunto.

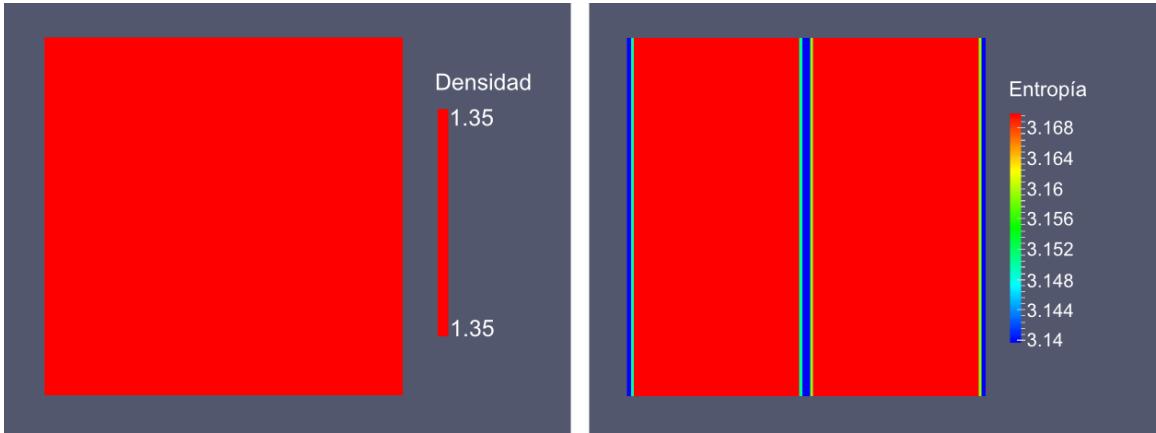


FIGURA 11 RESULTADOS DE DENSIDAD N_{ij} (IZQUIERDA) Y ENTROPÍA H_{ij} (DERECHA), PARA EL OPERADOR SUMA APLICADO A LA IMAGEN 2 DEL CONJUNTO DE PRUEBA

La Figura 11 muestra algunos de los resultados obtenidos para la imagen 2 del set de prueba. La regla utilizada en este caso para la construcción de la matriz M , es la Suma (Ecuación 9). Si bien la densidad (N_{ij}) no muestra signos de ninguna distinción particular, el resto de las variables calculadas (en este caso, entropía) muestran una clara detección de los bordes presentes en la imagen. Por otro lado, la diferencia de intensidades a ambos lados de la imagen, no se ve reflejada en los resultados. Esto es importante para la caracterización de texturas, ya que lo principal es distinguir entre formas y no entre intensidades.

Al realizar idénticas pruebas sobre la misma imagen, pero utilizando el Producto (Ecuación 10) como operador generador de M , los resultados fueron similares a los del operador Suma. Se observó una significativa distinción de los bordes y una clara independencia de la intensidad de los píxeles.

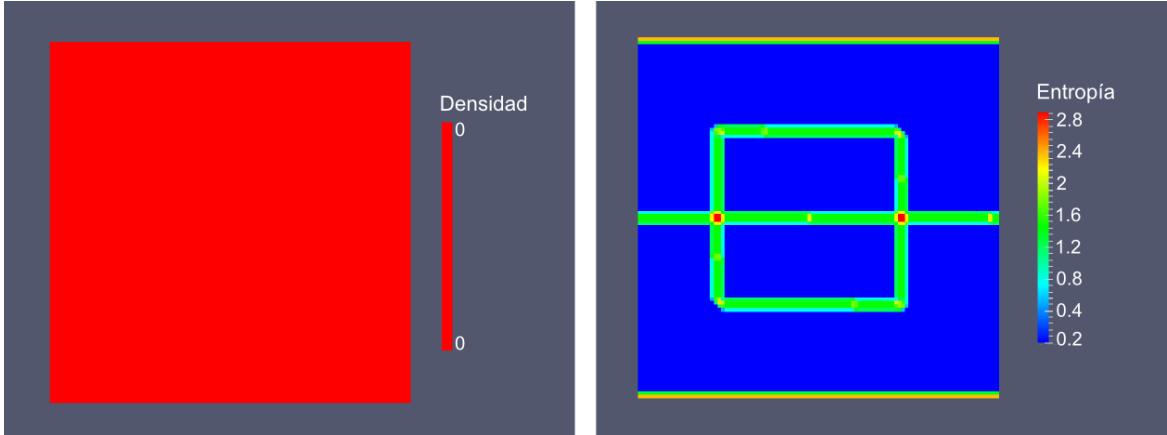


FIGURA 12 RESULTADOS DE DENSIDAD N_{ij} (IZQUIERDA) Y ENTROPÍA H_{ij} (DERECHA), PARA EL OPERADOR RESTA APLICADO A LA IMAGEN 3 DEL CONJUNTO DE PRUEBA.

De la misma manera, se aplicaron las reglas Suma y Producto a la imagen 3 del conjunto de prueba, obteniendo resultados similares. Al aplicar Resta (Ecuación 11), algunas de las variables resultado que antes identificaban bordes, ya no lo hicieron. No obstante, la entropía mantiene su comportamiento en comparación con el obtenido para otros operadores (Figura 12).

4.2 DISTRIBUCIÓN PROMEDIADA

Otro conjunto de pruebas fue conformado para verificar la real independencia del promedio de las intensidades en una textura y los datos retornados por el algoritmo. Esto es importante, ya que lo que es verdaderamente relevante para nuestro estudio es la distribución de los píxeles en una imagen y no el valor de su intensidad.

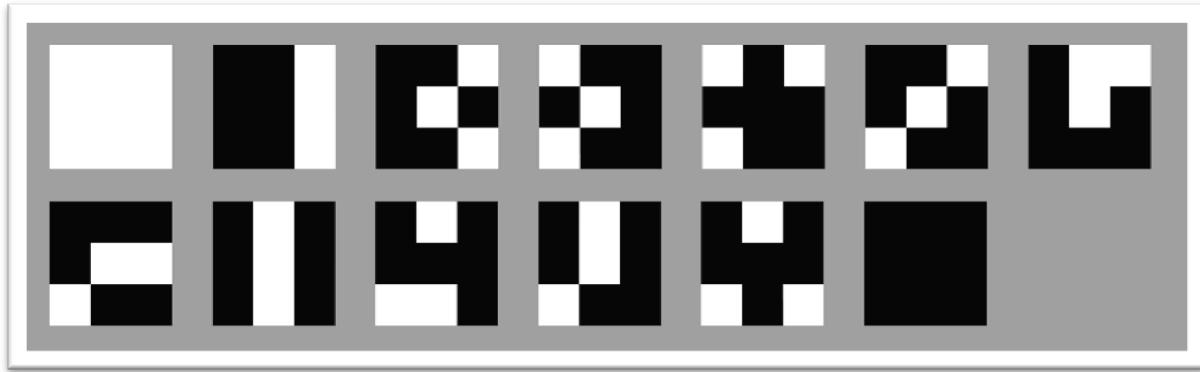


FIGURA 13 CONJUNTO DE PRUEBA. TODOS LOS ELEMENTOS, A EXCEPCIÓN DE LOS EXTREMOS, POSEEN 3 PÍXELES BLANCOS Y 6 NEGROS.

Para este estudio se trabajó con un conjunto de 13 imágenes pequeñas (3×3 píxeles) con igual promedio de intensidades pero diferente distribución espacial (Figura 13). Todas, a excepción de dos que son utilizadas como referencia (texturas planas blanca y negra), respetan esta propiedad.

Es importante remarcar el hecho de que, al encontrarnos trabajando con condiciones de bordes periódicas, procesar imágenes pequeñas que constituyan la base de patrones más grandes es equivalente a trabajar con esos patrones más grandes. Es decir, los bordes periódicos hacen que el patrón se repita en los laterales, para formar una textura mayor.

4.2.1 VALIDACIÓN EXPERIMENTAL

Se ejecutaron las pruebas utilizando los tres operadores (Suma, Producto, Resta), y se analizaron los promedios de las f_i para cada una de las imágenes luego de alcanzada la convergencia. Los dos elementos de referencia se comportaron según lo esperado, de forma constante y en el caso de la textura negra, igual a cero.

Si bien todos los operadores distinguieron entre las diferentes texturas, los resultados más interesantes se obtuvieron para el operador Resta. La Figura 14 muestra los promedios de las f_i para el operador Suma, donde se observa un comportamiento similar para casi todas las texturas del conjunto de prueba. Lamentablemente este comportamiento no es el que se desea que tenga el algoritmo, ya que como todos los patrones de prueba son diferentes, los resultados de la detección deben también resultar significativamente diferentes.

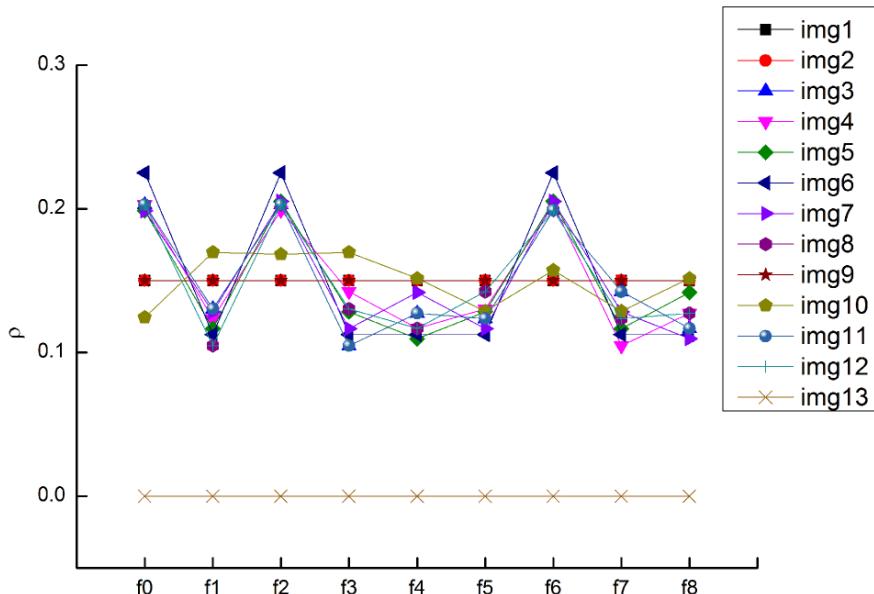


FIGURA 14 VALORES PROMEDIO DE LAS f_i PARA CADA IMAGEN DEL CONJUNTO DE PRUEBA. OPERADOR SUMA

Por otro lado, analizando la Figura 15 (operador Resta), es posible observar que el parámetro en estudio devuelve diferentes resultados para cada una de las imágenes. Los

conjuntos de f_i para cada textura se diferencian significativamente del resto, a pesar de que el promedio de las intensidades es el mismo en todos los casos.

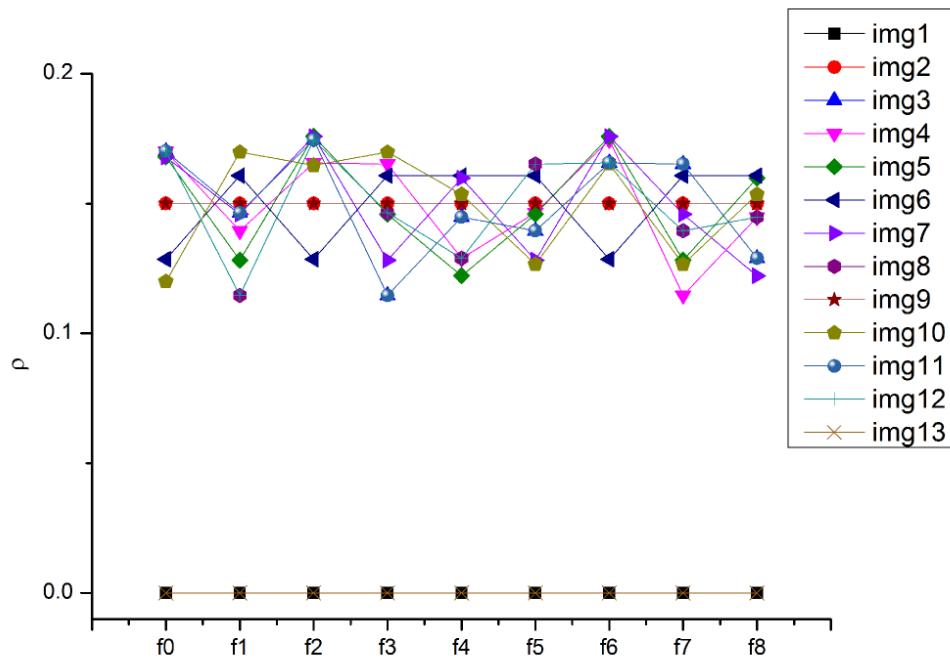


FIGURA 15 VALORES PROMEDIO DE LAS f_i PARA CADA IMAGEN DEL CONJUNTO DE PRUEBA. OPERADOR RESTA

La gráfica indica que el operador resta es significativamente más sensible, comparado con el operador Suma, a los cambios en la distribución de las intensidades en texturas como las del ejemplo.

4.3 SEPARACIÓN POR ÁNGULO

Uno de los parámetros que más nos interesa estudiar es el ángulo del vector velocidad en cada celda del *lattice*, descripto en la Ecuación 5 (e) como:

$$\text{Ángulo: } \theta_{ij} = \text{atan} \left(\frac{uy_{ij}}{ux_{ij}} \right), 0 \text{ si } ux_{ij} = 0$$

Dadas las características de la simulación, este ángulo puede ofrecer información relevante sobre la distribución de intensidades en la textura.

La prueba está constituida por dos etapas principales:

1. En primer lugar, se ejecuta el algoritmo para una imagen con presencia de solo dos texturas diferentes.
2. Se estudia el campo angular para la distribución f_i , para detectar diferencias entre ambas texturas.

La segunda etapa consiste en dividir en dos conjuntos el campo angular resultante de la ejecución del algoritmo. El dominio posible está entre 0 y 2π , y la división se realizará mediante el control de dos parámetros denominados α y β . El primero de los parámetros regula la posición de la línea de corte del círculo, y el segundo la apertura de los mismos (ver Figura 16). Alfa se define en el dominio $0 < \alpha < \pi$ y Beta en $0 < \beta < 1$, siendo 1 una apertura equivalente a π radianes.

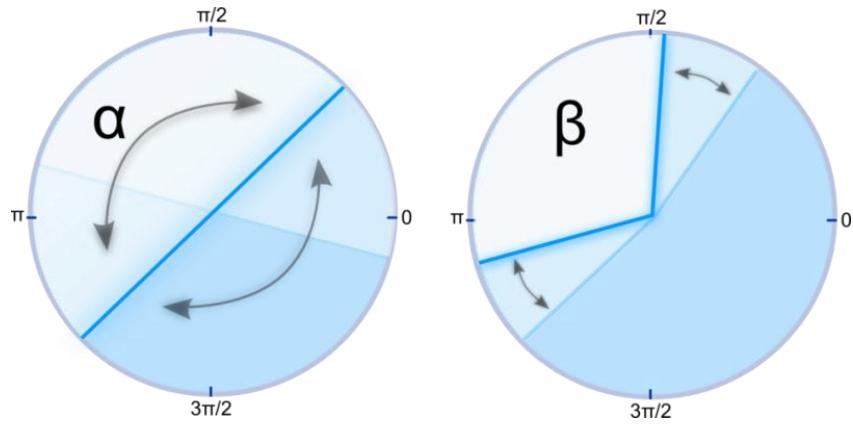


FIGURA 16 ESQUEMA DE SEPARACIÓN DEL ESPACIO ANGULAR PARA LOS PARÁMETROS α Y β

Se espera encontrar valores de α y β capaces de dejar en uno de los conjuntos, todos aquellos ángulos que pertenecen a una misma textura y en el otro, a los de la segunda textura. De esta forma, es posible observar si los ángulos del campo de velocidades son de importancia en el proceso de caracterización de texturas.

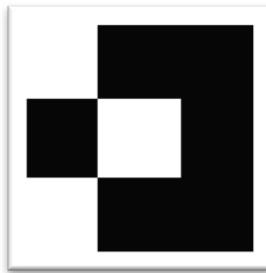


FIGURA 17 TEXTURA BASE DEL EXPERIMENTO

La textura elegida para el experimento es la que muestra la Figura 17. Pertenece al mismo conjunto detallado en la Sección 4.2. Es una textura simple, con solo dos intensidades: blanco y negro. Para poder incluir dos texturas en la misma imagen, trabajaremos ahora con dimensiones más grandes (90x90 píxeles).



FIGURA 18 TEXTURA BASE, MASCARA Y TEXTURA FINAL

Mediante la utilización de una máscara, se introdujo en la textura una rotación de ella misma, de manera que en la imagen quedarán representadas dos texturas diferentes. La Figura 18 muestra como a partir de una textura homogénea, se obtuvo la imagen compuesta.

4.3.1 VALIDACIÓN EXPERIMENTAL

La simulación ha sido ejecutada y los resultados (distribución de los ángulos) graficados mediante una herramienta que permitió realizar el ajuste manual de los parámetros α y β , hasta encontrar una buena configuración que separará la imagen en dos texturas claramente identificables. La aplicación en cuestión, toma la distribución de ángulos, resultante del algoritmo de detección, y los clasifica en dos, ubicándolos en el conjunto que correspondan (según α y β). Una nueva imagen se genera a partir de esa clasificación. Para cada ángulo, si pertenece a uno u otro conjunto, se pinta el pixel i, j al que representa en blanco o negro, respectivamente. El resultado, una imagen binaria que permite apreciar rápidamente, la distribución de ángulos resultante.

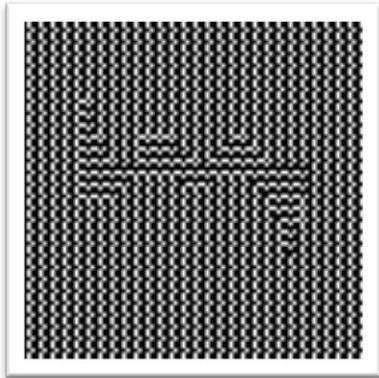


FIGURA 19 IMAGEN DE PRUEBA

Los resultados del algoritmo para la textura de prueba (Figura 19) muestran una distinción clara de las dos regiones. Es posible observar que no se ha obtenido una nueva imagen con una clara distinción entre ambas regiones, sino más bien una nueva figura compuesta por dos texturas (ver Figura 20). La ejecución fue realizada con el operador Resta (Ecuación 11).

La nueva textura generada tiene dos características interesantes. Por un lado, la textura próxima al contorno de la imagen, es completamente regular y no sufre perturbaciones. El núcleo de la imagen se ve completamente caótico y no hay una distribución constante de las intensidades.



FIGURA 20 DISTRIBUCIÓN DE ÁNGULOS PARA $\alpha = \frac{\pi}{6}$ Y $\beta = 0.56$

Surge, en este punto, la necesidad de homogeneizar la imagen, de manera que sea posible distinguir entre lo que es y no es parte de cada textura. Es posible lograr esto mediante la aplicación de un filtro de “smooth” que suavice la imagen, en combinación con un método de segmentación por umbralado (Shapiro, 2001).



FIGURA 21 RESULTADO DEL MÉTODO DE SEPARACIÓN DE ÁNGULOS CON FILTRO SMOOTH

Tal como es posible observar en la Figura 21, a pesar de que requiere de calibración manual, el algoritmo empleado es capaz de separar la imagen en dos regiones, diferenciando ambas texturas. En los experimentos siguientes se explora con más profundidad esta alternativa.

4.4 SEPARACIÓN POR ÁNGULO Y DIFUSIÓN

Dadas las características de las imágenes resultantes de los experimentos realizados con el algoritmo de separación por ángulo, resultó interesante combinar esa técnica con el concepto de difusión. La difusión anisotrópica es un concepto proveniente del campo de Procesamiento de Imágenes y Visión Computacional. Es también denominada “difusión Perona-Malik” y es una técnica que intenta reducir el ruido en una imagen, sin remover las partes importantes de ella, como bordes, líneas u otros detalles relevantes para la interpretación (Perona, 1987).

En el modelo de lattice-Boltzmann, la difusión consiste en la adición de un término constante en la construcción de la matriz M_{ij} , esto es:

$$\text{Suma: } M[i, j, k, n] = \frac{p_{ijk} + p_{ijn}}{\sum_{z=0}^l M[i, j, z, n]} + c_{ij}$$

ECUACIÓN 12 REGLA SUMA NORMALIZADA CON TÉRMINO DE DIFUSIÓN

$$\text{Producto: } M[i, j, k, n] = \frac{p_{ijk} * p_{ijn}}{\sum_{z=0}^l M[i, j, z, n]} + c_{ij}$$

ECUACIÓN 13 REGLA PRODUCTO NORMALIZADO CON TÉRMINO DE DIFUSIÓN

$$\text{Módulo - Resta: } M[i, j, k, n] = \frac{|p_{ijk} - p_{ijn}|}{\sum_{z=0}^l M[i, j, z, n]} + c_{ij}$$

ECUACIÓN 14 REGLA MÓDULO-RESTA NORMALIZADA CON TÉRMINO DE DIFUSIÓN

donde c_{ij} es un parámetro de difusión normalizado entre 0 y 1. La inclusión de este término modifica la matriz, de manera que ya no se encontrará normalizada al final de este proceso. Por esta razón, una nueva normalización debe ser realizada (Ecuación 15).

$$M[i, j, k, n] = \frac{M[i, j, k, n]}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 15 SEGUNDA NORMALIZACIÓN PARA LA DIFUSIÓN

Si bien el modelo de difusión admite la inclusión de diferentes valores de difusión en cada celda i, j , para este experimento se utilizó un mismo valor para toda la matriz c_{ij} . Esta característica es, en general, aprovechada por algoritmos que calculan c_{ij} como una función de la imagen, con el objetivo de realzar ciertas características (bordes, líneas, etc.). En este caso, esa información ya se encuentra presente en las ecuaciones de M_{ij} , por lo que no ha resultado de interés experimentar con esa característica.

4.4.1 VALIDACIÓN EXPERIMENTAL

Se ha utilizado para este experimento la misma imagen que en el experimento de la sección anterior (Figura 19). En primer lugar, se calculó el campo de ángulos para la simulación con el operador Resta, para valores de difusión iguales a 0%, 100%, 400% (Figura 22).

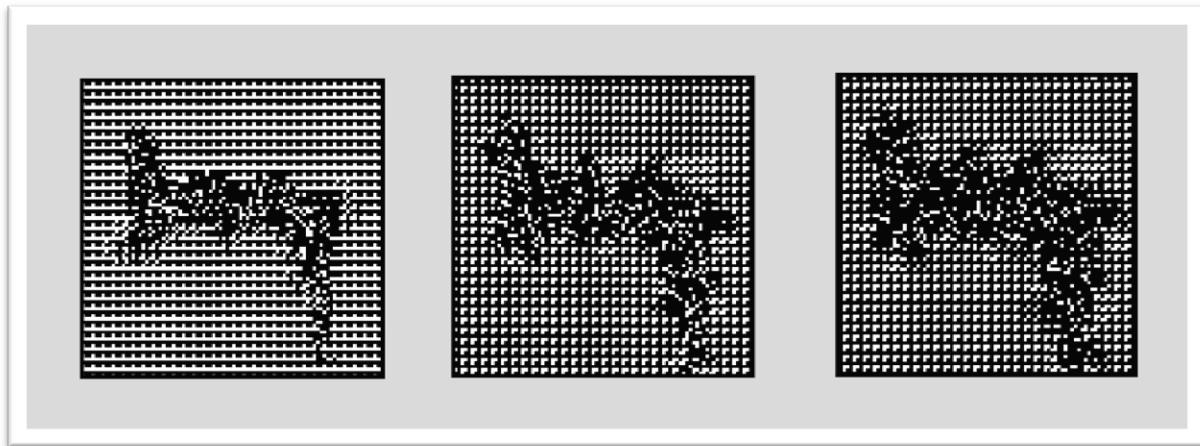


FIGURA 22 DISTRIBUCIÓN DE ÁNGULOS PARA $\alpha = \pi/6$ Y $\beta = 0.56$. VALORES DE DIFUSIÓN 0%, 100% Y 400% (DE IZQUIERDA A DERECHA, RESPECTIVAMENTE)

Se observó que a medida que el valor de difusión aumenta, hay una disminución de la definición de los bordes, es decir, una menor claridad en el umbral entre ambas texturas. La Figura 23 muestra en diferentes tonos de azul, el efecto del incremento de la difusión. El área más pequeña (azul oscuro) corresponde a la ausencia de difusión mientras que el área más grande (azul claro) pertenece a la segmentación obtenida para un valor de difusión igual al 400%.

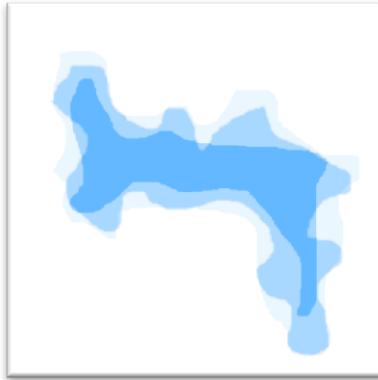


FIGURA 23 ESKEMA DE SUPERPOSICIÓN DE LA SEGMENTACIÓN PARA DIFUSIÓN DE 0%, 100% Y 400%

Para cuantificar estas diferencias se realizó un experimento utilizando el índice AOM (Abdel-Dayem, 2010; del Fresno 2009) y una segmentación de referencia (construida manualmente). La prueba consistió en ejecutar el algoritmo y calcular AOM para diferentes valores de difusión. La Figura 24 muestra los resultados obtenidos para porcentajes de difusión entre 0 y 200%. El mejor ajuste de la segmentación se logró para valores relativamente bajos de difusión (9,6% en este caso) y, en general, el algoritmo pierde rápidamente precisión a medida que se aumenta la difusión.

Por otro lado, si se observan los valores de ajuste calculados por el índice AOM, es posible notar que no resultan cercanos a 1, como debería ser en caso de que la segmentación sea óptima. Esto se debe a una falencia en la segmentación que proviene del método de umbralado. Si bien el umbral entre ambas texturas es correctamente detectado, la segmentación utilizada deja “huecos” (ver Figura 21) que AOM refleja como valores bajos de ajuste.

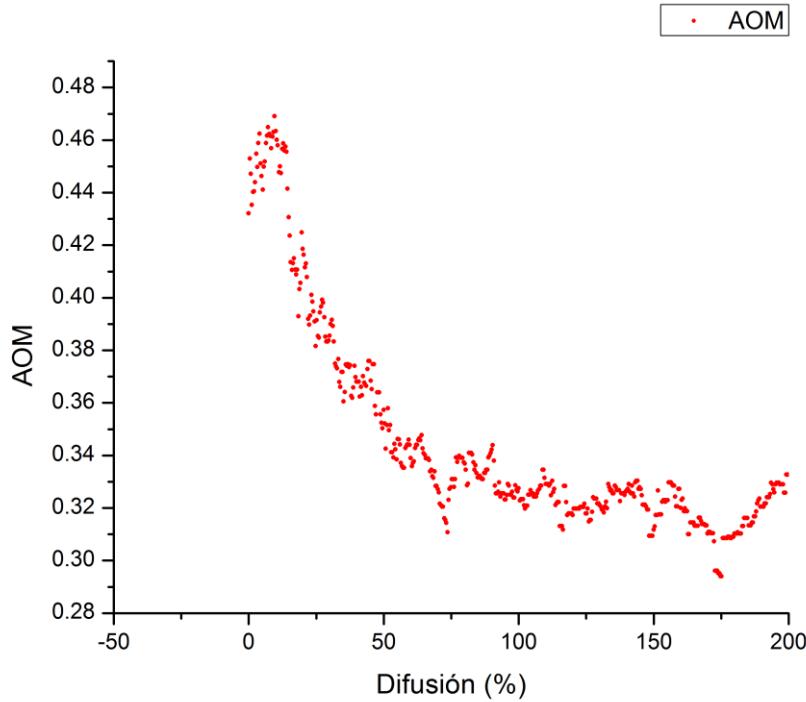


FIGURA 24 VALORES DE AOM PARA DIFERENTES PORCENTAJES DE DIFUSIÓN

A pesar de los valores bajos del índice AOM, el experimento confirma que el algoritmo es capaz de detectar diferencias entre texturas e identificarlas, más allá de los errores que se produzcan en la segmentación posterior.

4.5 CAMPO DE VELOCIDADES

Dado que el estudio de conjunto de ángulos ha demostrado ser de utilidad para la detección de texturas, se ha propuesto otro experimento que explore el campo de velocidades. La Ecuación 5 (e) postula que:

$$\text{Ángulo: } \theta_{ij} = \text{atan} \left(\frac{uy_{ij}}{ux_{ij}} \right), 0 \text{ si } ux_{ij} = 0$$

donde ux_{ij} y uy_{ij} son las velocidades en la dirección x e y , respectivamente, y se calculan según la Ecuación 5 (b) como:

$$\text{Velocidad: } \overrightarrow{u_{ij}} = \sum_{k=0}^l f_{ijk} \overrightarrow{e_k}$$

Esta relación entre el ángulo y las velocidades, vuelve atractivo el estudio de esta variable.

El experimento consiste en la construcción de gráficas cartesianas que representen el campo de velocidades ux, uy , para los operadores Suma y Resta en la construcción de la matriz M_{ij} . El conjunto de prueba incluye la imagen de textura compuesta de la Figura 19, y también la definida en la Figura 25.

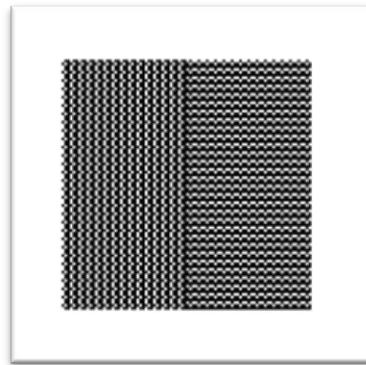


FIGURA 25 TEXTURA DE PRUEBA PARA EL CÁLCULO DEL CAMPO DE VELOCIDADES

Al igual que en el experimento anterior, se intenta estudiar también el efecto de la difusión en las gráficas de velocidad, para evaluar si tiene un efecto positivo en el proceso de detección de texturas.

4.5.1 VALIDACIÓN EXPERIMENTAL

La Figura 26 muestra la gráfica del espacio de velocidades para la Imagen de la Figura 19. Cada punto es la velocidad en un pixel. Pueden verse conjuntos de puntos agrupados, pero no hay una clara separación de los que componen una u otra textura. Sería deseable conseguir una separación clara de los conjuntos de puntos que haga posible la segmentación.

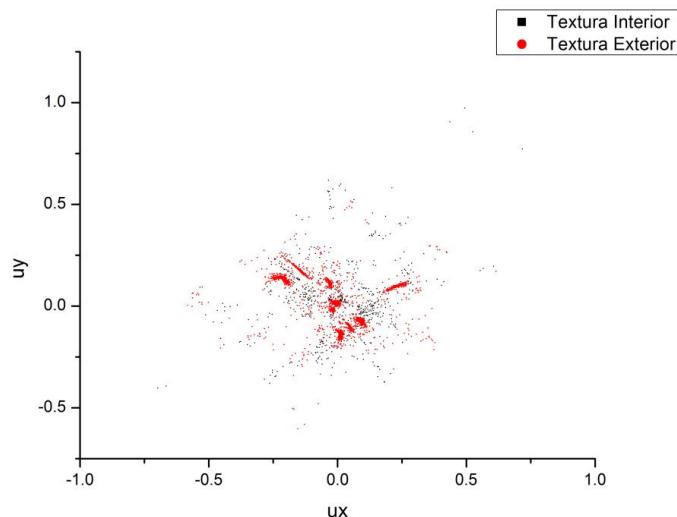


FIGURA 26 GRÁFICA UX,UY PARA EL OPERADOR RESTA. IMAGEN DE PRUEBA DE LA FIGURA 19

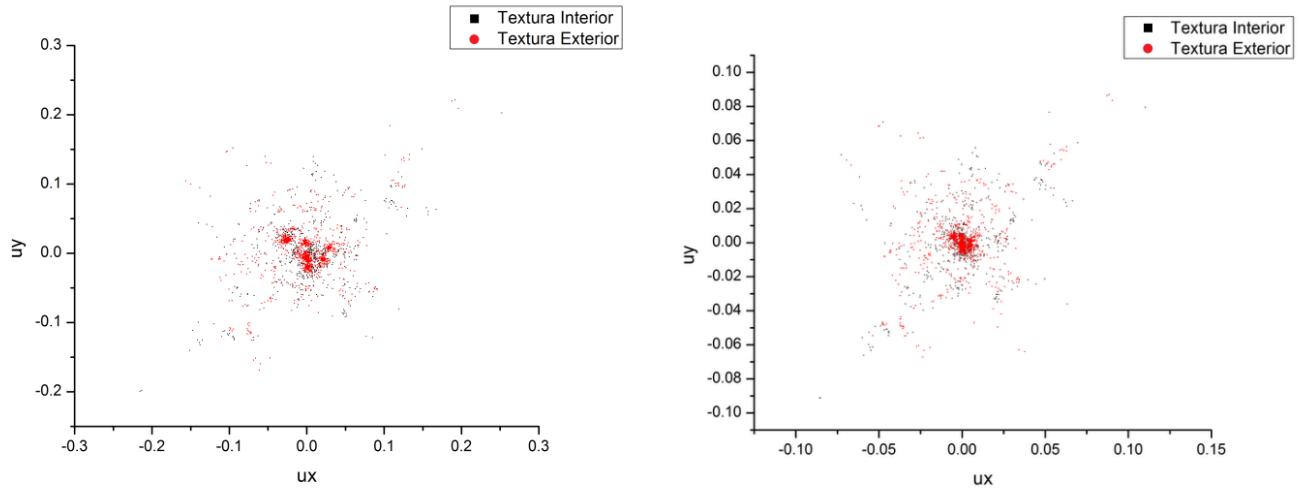


FIGURA 27 GRÁFICA UX,UY PARA EL OPERADOR RESTA. IMAGEN DE PRUEBA DE LA FIGURA 19. DIFUSIÓN=20% (IZQUIERDA) Y DIFUSIÓN=60% (DERECHA)

Con el motivo de detectar cuales puntos pertenecen a cada una de las texturas, se modificó la gráfica de manera que esta información estuviera visible en la figura. Mediante la ayuda de una segmentación manual de referencia, fue posible realizar una gráfica del campo de velocidades, coloreando los puntos de acuerdo a su pertenencia a la textura interior o exterior de la imagen (ver Figura 26).

Por otro lado, el efecto de la difusión puede apreciarse en la Figura 27. Puede observarse que la difusión no tiene un efecto positivo para la segmentación, ya que genera una tendencia hacia el 0 de las nubes de puntos en las gráficas de velocidad. Por esta razón, se descartan valores altos de difusión en las configuraciones de los experimentos subsiguientes

Del análisis de este gráfico se obtiene que, si bien las nubes de puntos no presentan una separación clara, sí están agrupadas de diferente manera, siendo la textura exterior la que se concentra en regiones y la interior la que se difumina de forma más homogénea.

El experimento fue repetido para la segunda textura de prueba (Figura 25), tanto para el operador Suma como para el operador Resta.

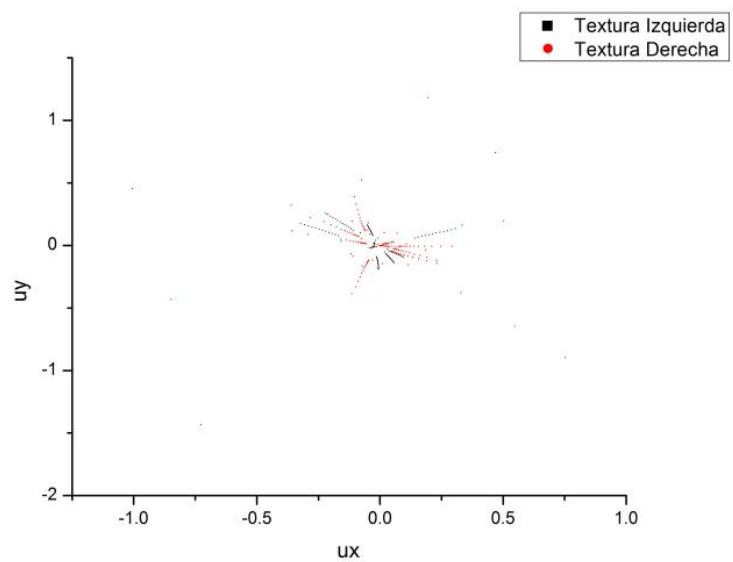


FIGURA 28 GRÁFICA UX,UY PARA EL OPERADOR RESTA. IMAGEN DE PRUEBA DE LA FIGURA 25

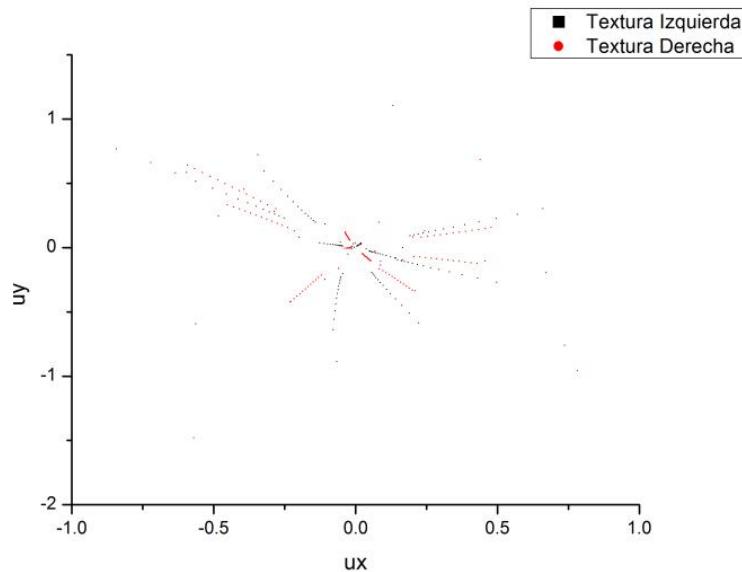


FIGURA 29 GRÁFICA UX,UY PARA EL OPERADOR SUMA. IMAGEN DE PRUEBA DE LA FIGURA 25

Ambas Figura 28 y Figura 29, muestran un comportamiento similar de las velocidades ante los operadores Suma y Resta. Los puntos se agrupan en rectas que poseen la misma pendiente. Notar que sobre una misma recta, es mínima la existencia de puntos de dos colores diferentes, confirmando que el ángulo de la velocidad es un parámetro importante de caracterización de las texturas.

4.6 ESPACIO FÁSICO

Adicionalmente a los análisis de la sección anterior, se han explorado otras alternativas que arrojen luz sobre las cuestiones relacionadas a la segmentación. Dado que el ángulo ha demostrado ser un buen indicador de las características de una textura, es posible combinarlo con las otras variables macroscópicas, para determinar si alguna de las configuraciones separa en regiones los datos devueltos por el algoritmo.

Los experimentos han sido realizados tanto como para la Suma como para la Resta. La Figura 30 muestra las 3 imágenes a ser utilizadas. Para cada una de ellas se explora la relación entre el ángulo del campo de velocidades, la densidad y la entropía del sistema.

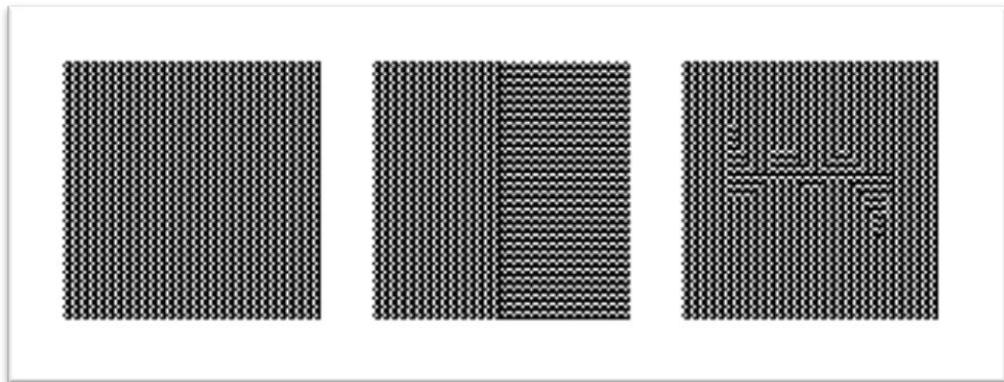


FIGURA 30 TEXTURA HOMOGÉNEA, COMPUESTA SIMPLE Y COMPUESTA IRREGULAR

4.6.1 VALIDACIÓN EXPERIMENTAL

ÁNGULO Y DENSIDAD

La Figura 31 es una gráfica en coordenadas polares del ángulo y la densidad para la imagen de textura homogénea. Todas las gráficas a continuación fueron obtenidas configurando el algoritmo para utilizar el operador Resta (Ecuación 11). Puede observarse que todos los pixeles se agrupan en un conjunto de 9 puntos. Estos nueve pares (*ángulo, densidad*) corresponden a cada una de las matrices diferentes existentes en la imagen.

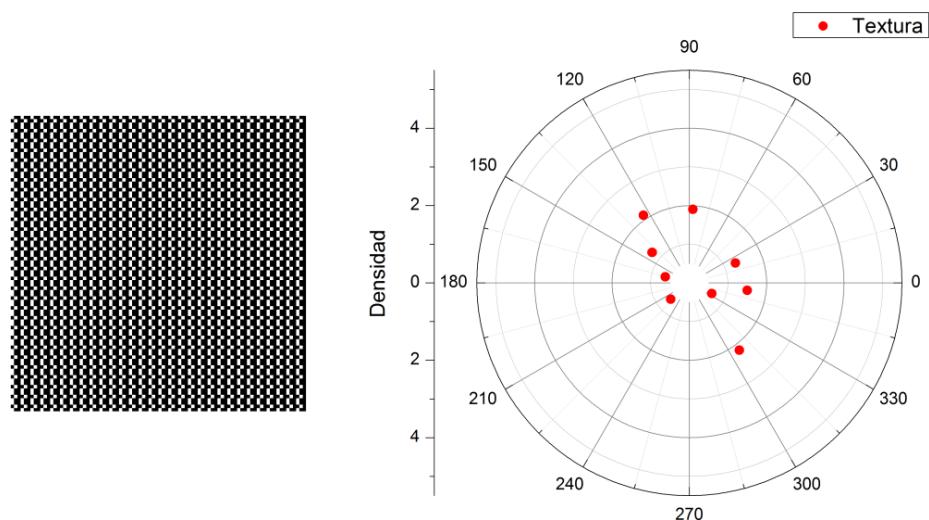


FIGURA 31 ÁNGULO Y DENSIDAD PARA UNA TEXTURA HOMOGÉNEA

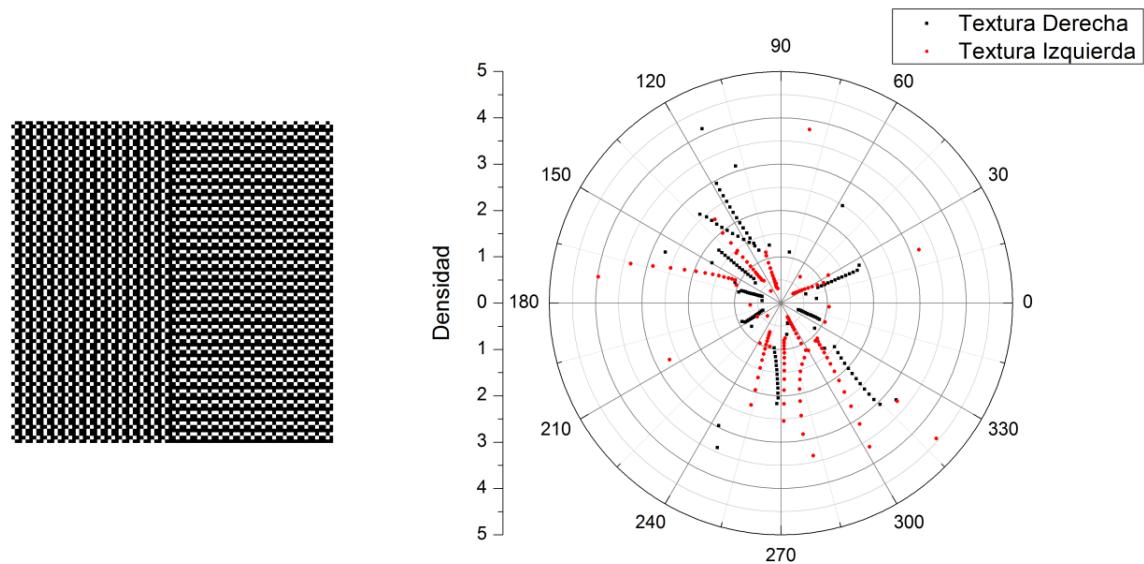


FIGURA 32 ÁNGULO Y DENSIDAD PARA UNA IMAGEN COMPUSETA SIMPLE

Al experimentar con imágenes compuestas por dos texturas y un borde simple, los resultados son los que muestra la Figura 32. Si bien no hay una separación marcada de las nubes de puntos, nuevamente, al igual que con las gráficas de velocidad, existe una disposición lineal guiada por el ángulo, que tiende a separar los puntos pertenecientes a una y a otra textura.

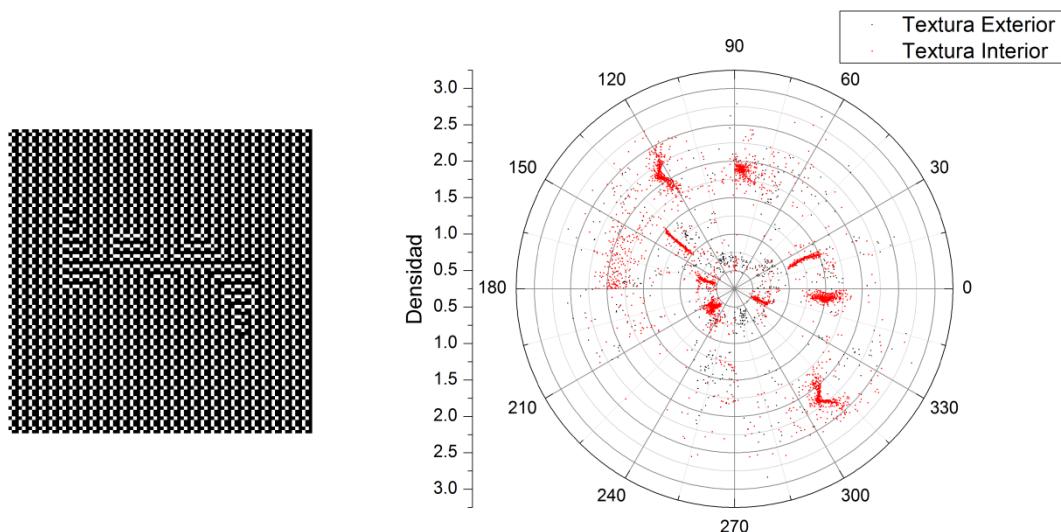


FIGURA 33 ÁNGULO Y DENSIDAD PARA UNA IMAGEN COMPUSTA

Al repetir los experimentos para la imagen de textura compuesta de la Figura 33, el comportamiento es similar al que posee la velocidad. Se pueden apreciar concentraciones distribuidas en el espacio angular, pertenecientes a la textura que compone el exterior.

Las pruebas fueron también realizadas para el operador Suma, y los resultados obtenidos fueron similares al de este operador. La Figura 34 muestra el ángulo y la densidad para la imagen de textura compuesta. Del análisis comparativo con los resultados obteni-

dos por el operador Suma (Figura 34) surge como conclusión que ambos se comportan de forma semejante.

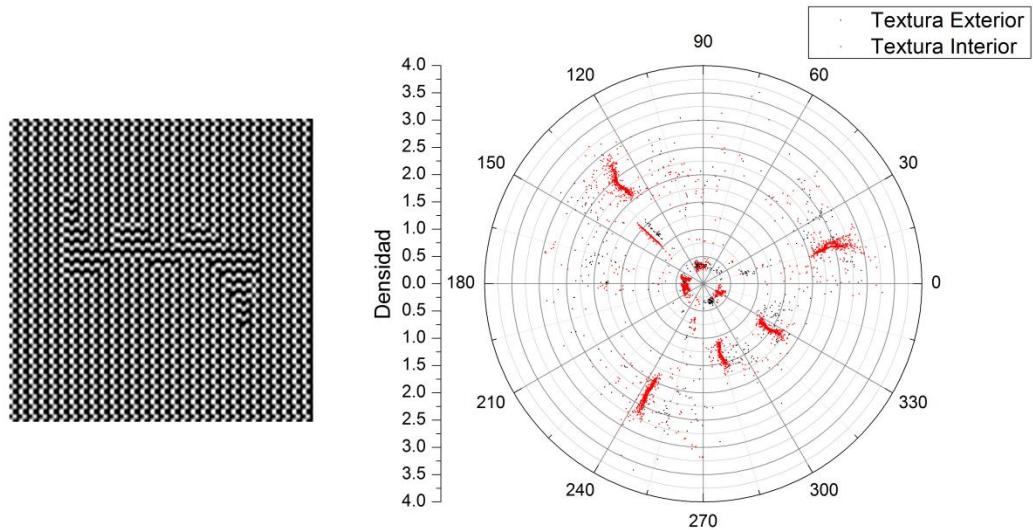


FIGURA 34 ÁNGULO Y DENSIDAD PARA UNA IMAGEN COMPUSTA (OPERADOR SUMA)

ÁNGULO Y ENTROPÍA

Las gráficas de los resultados para el ángulo y la entropía (Figura 35 y Figura 36) muestran, al igual que la densidad, que los puntos tienden a concentrarse en algunas regiones particulares, pero no es posible observar un patrón identifiable que pueda ayudar a distinguir si pertenecen o no a una textura, para realizar la detección y posterior segmentación de la imagen.

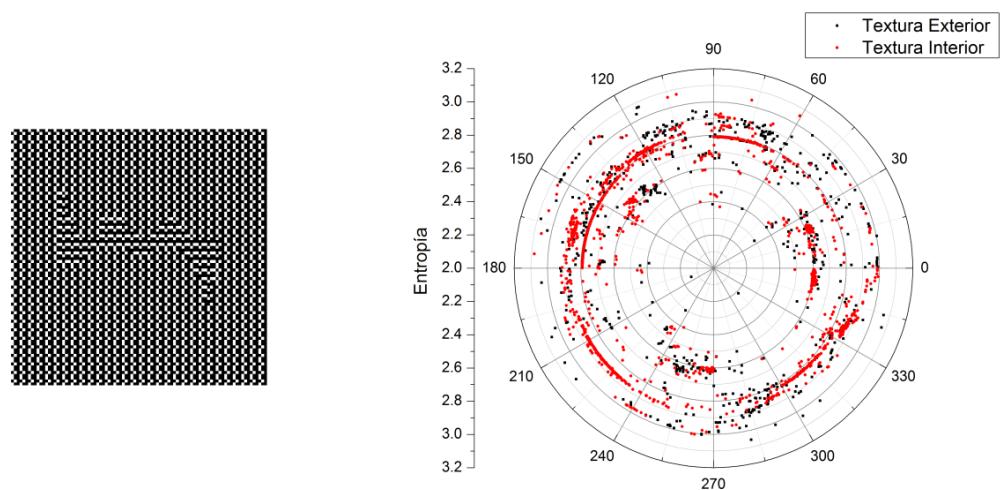


FIGURA 35 ÁNGULO Y ENTROPÍA PARA UNA IMAGEN COMPUSTA (OPERADOR RESTA)

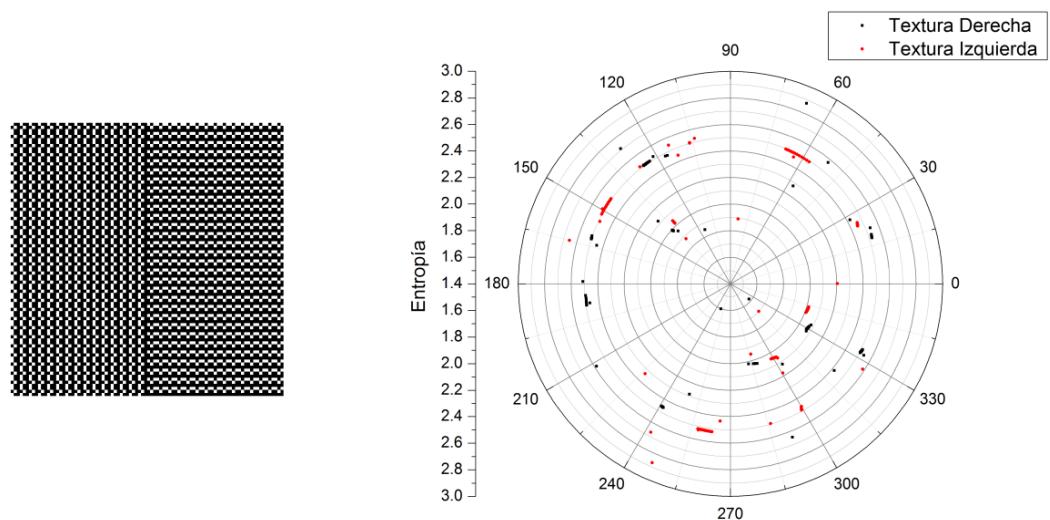


FIGURA 36 ÁNGULO Y ENTROPÍA PARA UNA IMAGEN COMPUSTA (OPERADOR SUMA)

4.7 DISCUSIÓN

Los experimentos realizados en esta sección muestran que, si bien hay algunos resultados prometedores, el algoritmo no se comporta según lo esperado. Básicamente, el objetivo buscado en este trabajo consiste en que, cuando la textura es homogénea y sin ruido, las gráficas de ángulo y densidad (o entropía) (ver Figura 31) presenten un único punto en el plano polar. Idealmente, en el tratamiento de figuras más complejas, deberían aparecer puntos por cada una de las texturas escondida en la imagen.

Se decidió entonces, investigar sobre las causas que dificultan la efectividad del método tal como fue presentado, comenzando por el estudio de texturas binarias homogéneas. La textura elegida (Figura 37) tiene como base el patrón binario de la Figura 13, que fue utilizado para el experimento de distribución promediada de la Sección 4.2.

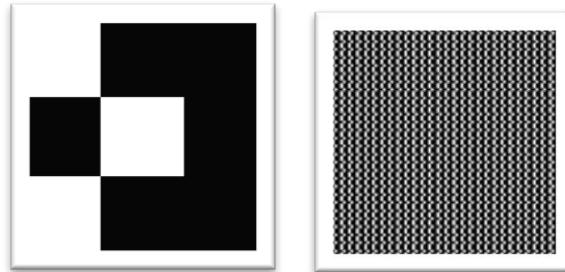


FIGURA 37 PATRÓN BASE (IZQUIERDA), IMAGEN DE TEXTURA HOMOGÉNEA (DERECHA)

Dicha textura, luego de ser tratada por nuestro algoritmo, mostraba una distribución de nueve puntos diferentes en el plano, que correspondían a cada una de las matrices que se generaban en el paso de construcción de la matriz de características del material M .

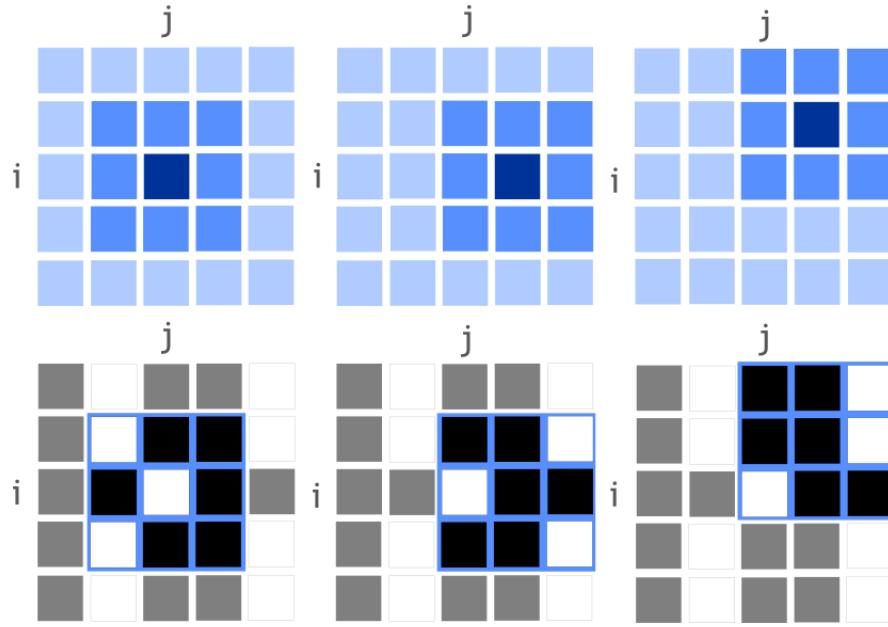


FIGURA 38 VENTANA DE 3x3 PARA TRES DE LOS VECINOS DEL PIXEL i,j

La Figura 38 muestra cómo la ventana utilizada para construir cada una de las matrices M_{ij} utiliza valores diferentes para el cálculo, como si cada una fuera una textura diferente. En el ejemplo sólo vemos tres de las nueve combinaciones diferentes, que rodean al pixel central, pero que en realidad son la misma textura homogénea. Para obtener un solo punto en la gráfica del ángulo y la densidad (o entropía) todos los píxeles de una textura deberían tener la misma matriz de transición. Esto significa que la falla del método está en la falta de homogeneización desde el primer paso del algoritmo, la creación de la matriz del medio.

5. GENERACIÓN DEL MEDIO

Para solucionar el problema planteado en la discusión de la sección anterior, se propuso una nueva manera de construir la matriz de transición, mediante el promedio entre las matrices de un mismo vecindario. Más precisamente, para un pixel i, j su M_{ij} (una vez que se ha inicializado con valores provenientes de la operación de Suma, Producto o Resta) se construye como:

$$M[i, j, k, n] = \frac{\sum_{m=-1}^1 \sum_{l=-1}^1 M[i + m, j + l, k, n]}{N}$$

ECUACIÓN 16 MATRIZ DE CARACTERÍSTICAS PROMEDIADA

donde N es la cantidad de vecinos del pixel i, j (9 en este caso). De esta forma, cada elemento de probabilidad de transición de una dirección k a una dirección n es, en realidad, un promedio de esas probabilidades en la vecindad del pixel i, j . Como consecuencia directa de este proceso, la matriz M para una textura homogénea es la misma para todos los píxeles.

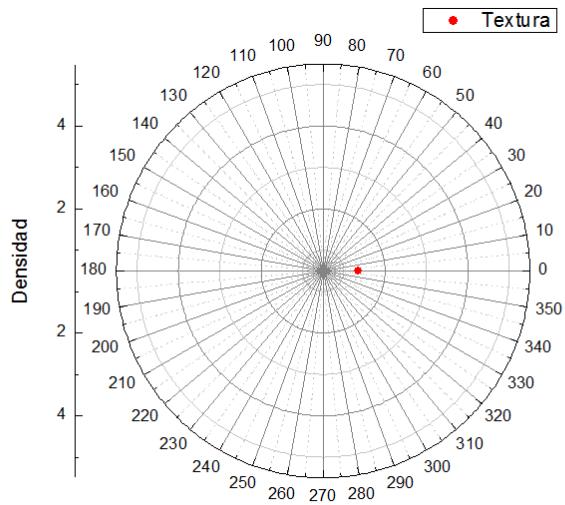


FIGURA 39 ÁNGULO Y DENSIDAD PARA UNA TEXTURA HOMOGÉNEA

La Figura 39 muestra los resultados de la aplicación del algoritmo a una textura homogénea, con la matriz M construida con el vecindario promediado. A diferencia de los casos homogéneos anteriores, en este se puede observar que todos los píxeles se acumulan en un solo punto de la gráfica.

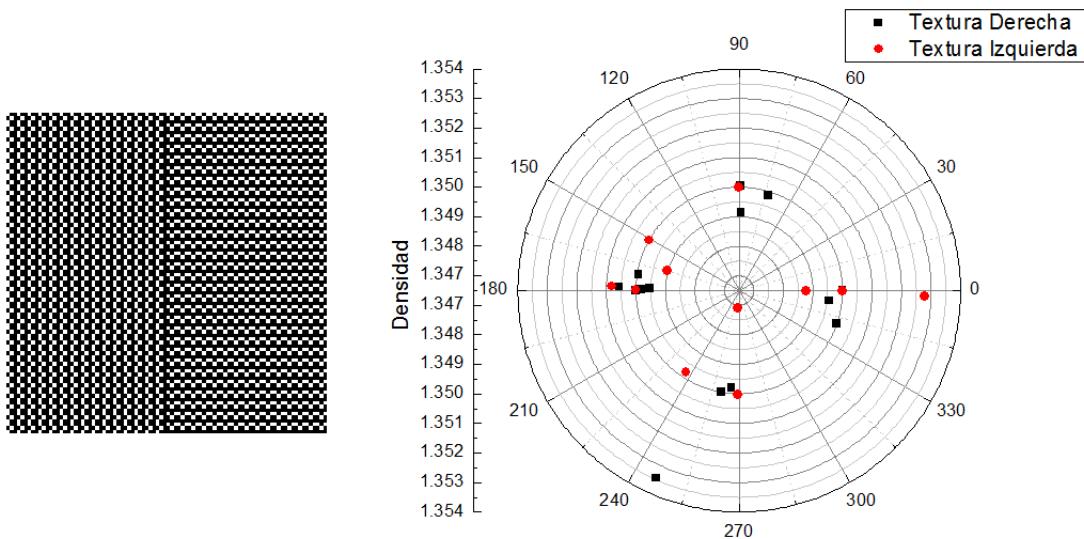


FIGURA 40 RESULTADOS DEL ALGORITMO CON VECINDARIO PROMEDIADO PARA UNA TEXTURA COMPUSTA

Al experimentar con texturas compuestas (Figura 40), no fue posible distinguir entre las dos texturas. Si bien para cada una de las texturas homogéneas se generan los mismos valores en la matriz M , al observar los resultados, notamos una dispersión de los puntos, y no dos concentraciones claramente diferenciables, como se esperaba. Al investigar las concentraciones para cada uno de los ángulos se observó que, tanto para la textura homogénea como para la compuesta, casi la totalidad de los valores se encontraban cercanos al ángulo 0 (Figura 41).

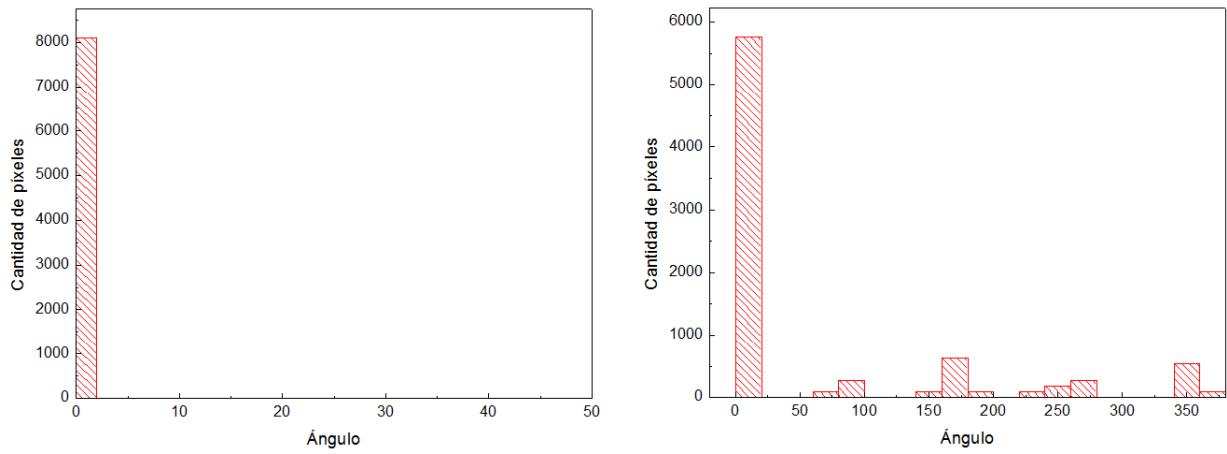


FIGURA 41 HISTOGRAMA PARA LA TEXTURA HOMOGÉNEA (IZQUIERDA) Y PARA LA COMPUSTA (DERECHA)

El experimento se repitió para otras texturas, con diferentes configuraciones de intensidades, obteniendo siempre los mismos resultados, una alta concentración de valores en 0.

En resumen, se ha conseguido una forma de garantizar que texturas homogéneas generen matrices de características homogéneas, pero al realizar la simulación, sólo obtenemos datos en cero. Para entender por qué sucede esto, es preciso volver a la definición que ofrece la Ecuación 5 (e) acerca del ángulo:

$$\text{Ángulo: } \theta_{ij} = \text{atan} \left(\frac{uy_{ij}}{ux_{ij}} \right), \text{ 0 si } ux_{ij} = 0$$

En todos los experimentos con diferentes texturas, observamos que el ángulo era igual a 0, a causa de que la velocidad (ux_{ij}) era también nula.

5.1 ESTUDIO DE LA PROBLEMÁTICA EN 1D

Para entender las causas de este comportamiento, se trabajó con un modelo lattice-Boltzmann simplificado. En lugar de trabajar con imágenes bidimensionales, se optó por estudiar el modelo unidimensional, lo que reduce significativamente la complejidad del método. En vez de una grilla bidimensional de tamaño $x \times y$, se cuenta con una ristra de celdas, cada una con sus correspondientes vectores de velocidad (Figura 42).

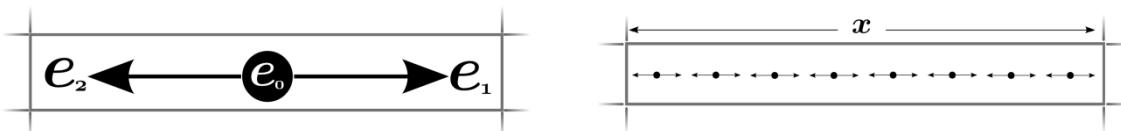


FIGURA 42 VECTORES DE DIRECCIÓN PARA UN ESQUEMA UNIDIMENSIONAL DE 2 VECINOS (IZQUIERDA). REPRESENTACIÓN DEL LÁTICE (DERECHA).

A modo de ejemplo se utilizó, para el experimento en 1D, una “textura” homogénea compuesta por el patrón 2-4-3. Se fijaron condiciones de contorno periódicas, es decir que ambos extremos del vector de datos se consideran “conectados”.

La textura en una dimensión queda definida como:

$$2 - 4 - 3 - 2 - 4 - 3 - 2 - 4 - 3 - 2 - 4 - 3 - 2 - 4 - 3 - \dots$$

Para este caso, construimos el vector M de características usando una nueva regla, la regla “División”, definida como:

$$\text{División: } M[i, k, n] = \frac{p_{ik}/(p_{in} + 1)}{\sum_{z=0}^l M[i, z, n]}$$

ECUACIÓN 17 REGLA DIVISIÓN 1D NORMALIZADA PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

El término 1 en el denominador evita la división por 0. Una propiedad interesante de este operador, es que a diferencia de la suma y del módulo de la resta, genera matrices no simétricas, es decir:

$$p_{ik}/(p_{in} + 1) \neq p_{in}/(p_{ik} + 1)$$

La Figura 43 muestra las tres matrices que se obtienen en vecindarios de tres píxeles, según el pixel central sea 2, 4 o 3.

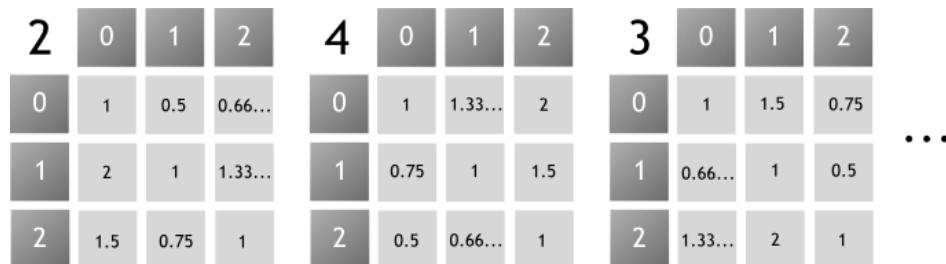


FIGURA 43 MATRICES DE TEXTURA PARA EL OPERADOR DIVISIÓN EN 1D

Luego, se aplica el promediado por vecindario, de forma que obtenemos:

2	0	1	2	4	0	1	2	3	0	1	2	• • •
0	1	1.11...	1.13	0	1	1.11...	1.13	0	1	1.11...	1.13	
1	1.13	1	1.11...	1	1.13	1	1.11...	1	1.13	1	1.11...	
2	1.11...	0.13	1	2	1.11...	0.13	1	2	1.11...	0.13	1	

FIGURA 44 MATRICES DE TEXTURA LUEGO DE APLICAR EL PROMEDIADO POR VECINDARIO

La Figura 44 muestra que, al ser una textura homogénea, todas las matrices poseen la misma distribución. Finalmente, se normalizan las columnas, para obtener las matrices estocásticas necesarias para asegurar la convergencia del método (Figura 45).

2	0	1	2	4	0	1	2	3	0	1	2	• • •
0	0.30	0.34	0.35	0	0.30	0.34	0.35	0	0.30	0.34	0.35	
1	0.35	0.30	0.34	1	0.35	0.30	0.34	1	0.35	0.30	0.34	
2	0.34	0.35	0.30	2	0.34	0.35	0.30	2	0.34	0.35	0.30	

FIGURA 45 MATRICES DE TEXTURA LUEGO DE LA NORMALIZACIÓN

Calculamos ahora algunos pasos de iteración de la simulación para observar el comportamiento de la distribución de fotones f . Recordando, la Ecuación 4 postula que:

$$f_i(\vec{x} + \vec{e}_l, t + 1) = \sum_j M_{ij} f_j(\vec{x}, t), i = 0, 1, \dots, l$$

Tomaremos entonces la celda 1 del *látice*, a manera de ejemplo y para simplificar el estudio, ya que al ser una textura homogénea, todas las celdas se comportan de la misma manera.

En el tiempo t , se tiene que:

$$f_1(0, t) = 0.15$$

$$f_1(1, t) = 0.15$$

$$f_1(2, t) = 0.15$$

El valor elegido para la inicialización del *látice* es, como se puede observar, 0.15. Luego, el primer paso de iteración ($t + 1$),

$$f_1(0, t + 1) = M_1 f_1$$

$$f_1(1, t + 1) = M_0 f_0$$

$$f_1(2, t + 1) = M_2 f_2$$

Así,

$$f_1(0, t + 1) = (0.15 \cdot 0.30) + (0.15 \cdot 0.34) + (0.15 \cdot 0.35) = 0.15$$

$$f_1(1, t + 1) = (0.15 \cdot 0.35) + (0.15 \cdot 0.30) + (0.15 \cdot 0.34) = 0.15$$

$$f_1(2, t + 1) = (0.15 \cdot 0.34) + (0.15 \cdot 0.35) + (0.15 \cdot 0.30) = 0.15$$

Al ser $f(t) = f(t + 1)$ el algoritmo converge en el primer paso de iteración. Y no solo eso, sino que además, todas las f_i contienen los mismos valores para todas las fuerzas, ya que esto mismo sucedió en todas las celdas del *látice*.

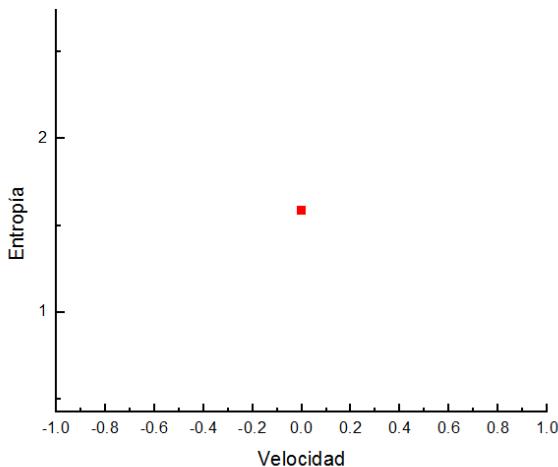


FIGURA 46 GRÁFICA DE LA VELOCIDAD Y LA ENTROPÍA PARA EL CASO 1D HOMOGÉNEO

Al realizar la gráfica de entropía vs velocidad (en 1D utilizamos la velocidad en ausencia de ángulo), vemos que el balance de fuerzas en cada celda (todas contienen 0.15 al finalizar la simulación) hace que la velocidad sea cero (Figura 46). Esto es lo mismo que ocurre en el caso 2D estudiado en la sección anterior.

La razón de este comportamiento es la simetría en las matrices de transición cuando se aplica el método del vecindario promediado. La Figura 44 muestra la configuración de valores de dicha matriz, luego de realizar el promedio con su vecindario. Si observamos detenidamente, la fila 0 y la columna 0 suman lo mismo, así como también la fila 1 y la columna 1, la fila 2 y la columna 2. Esta simetría, entonces, se traslada al paso de normalización por columnas (Figura 45). Al normalizar las columnas, se normalizan también de manera implícita, las filas. De esta forma, cuando se realiza la simulación, lo único que se hace es multiplicar el valor de inicialización del látice por la unidad.

Detectada la raíz del problema, es posible entonces solucionarlo de diferentes maneras. Básicamente, la idea es que las filas de cada una de las matrices que componen M no se encuentren normalizadas. Una alternativa viable consiste en ponderar cada una de las matrices que conforman el vecindario en el primer paso de construcción, antes del promediado y la normalización; por ejemplo, seleccionar el “píxel” central de la ventana de estudio como factor de ponderación. De esta manera, para la división:

$$\text{División: } M[i, k, n] = \frac{(p_{ik}/(p_{in} + 1)) \cdot p_0}{\sum_{z=0}^l M[i, z, n]}$$

ECUACIÓN 18 OPERADOR DIVISIÓN 1D NORMALIZADA CON PONDERACIÓN POR PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

donde p_0 es el píxel central de la ventana de generación (vecindario). Agregando este factor, la matriz M para el operador División viene dada como muestra la Figura 47.

2	0	1	2	4	0	1	2	3	0	1	2	• • •
0	2	1	1.33...	0	4	5.33...	8	0	3	4.5	2.25	
1	4	2	2.66...	1	3	4	6	1	2	3	1.5	
2	3	1.5	2	2	2	2.66...	4	2	4	6	3	

FIGURA 47 MATRICES PARA LA TEXTURA HOMOGÉNEA, UTILIZANDO LA PONDERACIÓN CON EL PIXEL CENTRAL

En la Figura 47, la primera de las matrices fue ponderada utilizando el valor “2”, la segunda el “4”, y la tercera el “3”. Promediando (Figura 48):

2	0	1	2	4	0	1	2	3	0	1	2
0	3	3.61...	3.86...	0	3	3.61...	3.86...	0	3	3.61...	3.86...
1	3	3	3.38...	1	3	3	3.38...	1	3	3	3.38...
2	3	3.38	3	2	3	3.38	3	2	3	3.38	3

FIGURA 48 MATRICES DE CARACTERÍSTICAS, LUEGO DE APLICAR EL PROMEDIADO POR VECINDARIO

Y finalmente, el paso de normalización (Figura 49):

2	0	1	2	4	0	1	2	3	0	1	2
0	0.33	0.36	0.37	0	0.33	0.36	0.37	0	0.33	0.36	0.37
1	0.33	0.30	0.33	1	0.33	0.30	0.33	1	0.33	0.30	0.33
2	0.33	0.33	0.29	2	0.33	0.33	0.29	2	0.33	0.33	0.29

FIGURA 49 MATRICES DE CARACTERÍSTICAS NORMALIZADAS

En la Figura 49 puede observarse que las filas no se encuentran normalizadas, habiendo corregido entonces el problema que llevaba a 0 las velocidades luego de la simulación.

5.2 ALGORITMO PARA LA DETECCIÓN DE TEXTURAS EN 1D

Para realizar la detección y segmentación en una dimensión, utilizando el modelo de iluminación difusa propuesto en este trabajo, es preciso seguir el siguiente esquema:

1. Inicialización de la matriz de características con ponderación por píxel central (p_0) mediante alguna de las técnicas propuestas en las siguientes ecuaciones (Ecuación 18 a Ecuación 21):

$$\text{Suma: } M[i, k, n] = \frac{(p_{ik} + p_{in}) \cdot p_0}{\sum_{z=0}^l M[i, z, n]}$$

ECUACIÓN 19 OPERADOR SUMA 1D NORMALIZADO CON PONDERACIÓN POR PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Producto: } M[i, k, n] = \frac{(p_{ik} * p_{in}) \cdot p_0}{\sum_{z=0}^l M[i, z, n]}$$

ECUACIÓN 20 OPERADOR PRODUCTO 1D NORMALIZADO CON PONDERACIÓN POR PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Módulo - Resta: } M[i, k, n] = \frac{(|p_{ik} - p_{in}|) \cdot p_0}{\sum_{z=0}^l M[i, z, n]}$$

ECUACIÓN 21 OPERADOR MODULO-RESTA 1D NORMALIZADO CON PONDERACIÓN POR PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{División: } M[i, k, n] = \frac{(p_{ik}/(p_{in} + 1)) \cdot p_0}{\sum_{z=0}^l M[i, z, n]}$$

2. Aplicación del promediado por vecindario, propuesto en la Ecuación 16, que homogeniza las matrices que son parte de una misma textura. Dicha ecuación, para un vecindario generalizado de tamaño L , viene dada por:

$$M[i, k, n] = \frac{\sum_{m=-\frac{L}{2}}^{\frac{L}{2}} M[i + l, k, n]}{L}$$

ECUACIÓN 22 MATRIZ DE CARACTERÍSTICAS 1D PROMEDIADA GENERALIZADA PARA VECINDARIOS DE TAMAÑO L

3. Se calculan luego, los pasos de iteración mediante simulación computacional del comportamiento de la distribución f de fotones. Para esto, se sigue la Ecuación 4, que postula:

$$f_i(\vec{x} + \vec{e}_l, t + 1) = \sum_j M_{ij} f_j(\vec{x}, t), i = 0, 1, \dots, l$$

Se itera la ecuación hasta que $f(t) = f(t + 1)$, momento en el que se ha alcanzado la convergencia del sistema.

4. Se extrae información a partir de la distribución f resultante. Ante la ausencia de más dimensiones que permitan calcular ángulos, la velocidad concentra toda la información sobre la composición de la textura. Dicha variable macroscópica viene dada por la Ecuación 5 (b):

$$\text{Velocidad: } \vec{u}_l = \sum_{k=0}^l f_{ik} \vec{e}_k$$

5. Existe una relación entre las características de las texturas y los valores de velocidad obtenidos para esos píxeles de la imagen. Es decir, texturas parecidas generan valores de velocidad parecidos.
6. A partir del campo de velocidades obtenido en el paso anterior, es posible realizar una segmentación por umbralado. Esta segmentación se realiza tal como fue explicada en la Sección 4.3 y puede ser fácilmente identificada mediante la utilización de un histograma. El campo de velocidades resultante de la ejecución del algoritmo es dividido de manera que aquellos rangos de velocidad con más cantidad de elementos (en el histograma), son considerados como parte de una misma textura.

Una vez establecido el método en una dimensión, es posible retomar los experimentos para corroborarlo (tal como se realizó en la Sección 4). Para los experimentos de las siguientes secciones se consideró el operador “División” definido en la Ecuación 18, además de la ponderación por el pixel central, que evita que tanto ángulos como velocidades queden en valores muy cercanos a 0.

5.3 CASO DE ESTUDIO: VECINDARIO PROMEDIADO CON PONDERACIÓN

Para estudiar el correcto funcionamiento del método en una dimensión, se creó un nuevo caso de estudio, compuesto por tres texturas diferentes, y en diferentes concentraciones. Los patrones elegidos fueron:

189 – 43 – 103

70 – 153 – 23

45 – 23 – 120

Concentradas con un 22%, 33% y 45% respectivamente, en tres regiones diferentes de la ristra de datos. Se utilizó nuevamente el operador “División”, el vecindario promediado y ponderación con el valor central del vecindario.

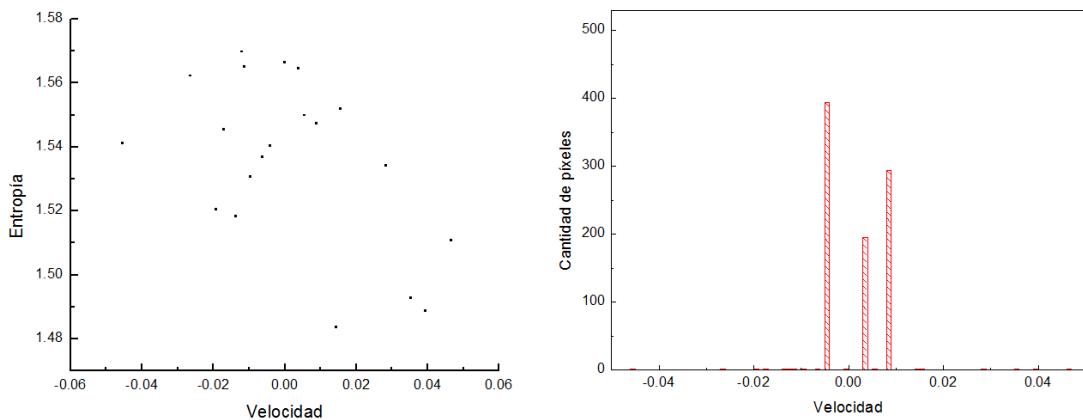


FIGURA 50 VELOCIDAD Y ENTROPIA PARA EL CASO DE ESTUDIO (IZQUIERDA) E HISTOGRAMA DE VELOCIDADES (DERECHA)

La Figura 50 muestra, por un lado, la gráfica de entropía vs velocidades, y por otro, el histograma para diferentes valores de velocidad. Si bien analizando la primera gráfica no

puede apreciarse una diferenciación clara de las tres texturas, al visualizar el histograma, vemos como claramente se encuentran presentes tres concentraciones de velocidad, y algunas otras dispersas que constituyen los bordes entre las tres texturas.

Efectivamente, en este caso, el algoritmo aquí presentado es capaz de hacer una detección y diferenciación de las texturas presentes en una imagen unidimensional (que también puede considerarse una señal discreta). Es deseable también, verificar la robustez del método ante la presencia de ruido. Se repitieron los experimentos, para la misma configuración de texturas, pero con ruidos del 1% y del 10%.

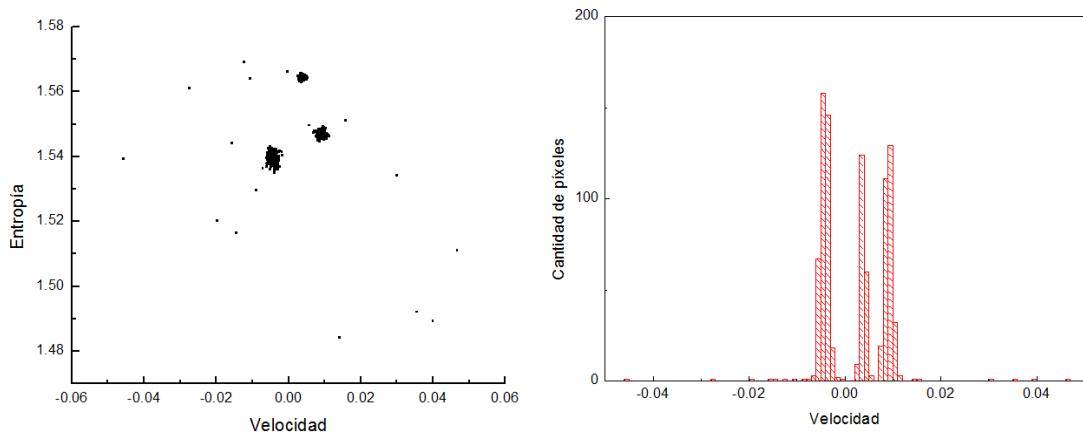


FIGURA 51 VELOCIDAD Y ENTROPIA (IZQUIERDA) E HISTOGRAMA DE VELOCIDADES (DERECHA) PARA RUIDO = 1%

En la Figura 51 se observan los efectos del ruido en la detección de las texturas. A diferencia de lo que se aprecia en la gráfica de entropía vs velocidad de la Figura 50, en esta última figura, pueden apreciarse concentraciones de puntos alrededor de ciertas regiones que determinan las texturas.

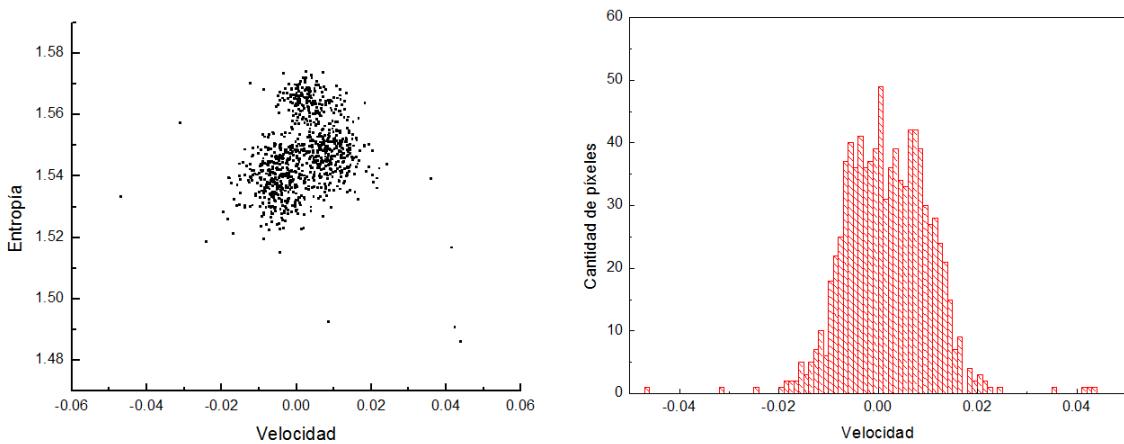


FIGURA 52 VELOCIDAD Y ENTROPIA (IZQUIERDA) E HISTOGRAMA DE VELOCIDADES (DERECHA) PARA RUIDO = 10%

Finalmente, la Figura 52 muestra la respuesta del algoritmo ante un ruido del 10%. Los conjuntos de puntos en la gráfica entropía vs velocidad ya no se encuentran claramente diferenciados, y el histograma confirma este hecho.

5.4 ESTUDIO DE LA PROBLEMÁTICA EN 2D

5.5 FORMALIZACIÓN DEL MÉTODO

Para realizar la detección y segmentación en dos dimensiones utilizando el modelo de iluminación difusa propuesto en este trabajo, es preciso seguir el siguiente esquema:

1. Inicialización de la matriz de características con ponderación por píxel central (p_0) (o algún otro criterio de ponderación) mediante alguna de las técnicas propuestas en las siguientes ecuaciones:

$$\text{Suma: } M[i, j, k, n] = \frac{(p_{ijk} + p_{ijn}) \cdot p_0}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 23 OPERADOR SUMA 2D NORMALIZADO CON PONDERACIÓN POR PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Producto: } M[i, j, k, n] = \frac{(p_{ijk} * p_{ijn}) \cdot p_0}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 24 OPERADOR PRODUCTO 2D NORMALIZADO CON PONDERACIÓN POR PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{Módulo - Resta: } M[i, j, k, n] = \frac{(|p_{ijk} - p_{ijn}|) \cdot p_0}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 25 OPERADOR MÓDULO-RESTA 2D NORMALIZADO CON PONDERACIÓN POR PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

$$\text{División: } M[i, j, k, n] = \frac{(p_{ik}/(p_{in} + 1)) \cdot p_0}{\sum_{z=0}^l M[i, j, z, n]}$$

ECUACIÓN 26 OPERADOR DIVISIÓN 2D NORMALIZADO CON PONDERACIÓN DE PIXEL CENTRAL PARA LA CONSTRUCCIÓN DE LA MATRIZ DE CARACTERÍSTICAS

2. Luego de esta fase de inicialización, es posible modificar la matriz de manera que algunas de las características de la imagen sean resaltadas. Las alternativas estudiadas fueron:

Promediado por vecindario: Su propiedad es homogeneizar las matrices que son parte de un mismo vecindario. Áreas con la misma textura, poseerán misma matriz en promedio. La ecuación, para un vecindario generalizado de tamaño $S \times L$, viene dada por:

$$M[i, j, k, n] = \frac{\sum_{s=-\frac{S}{2}}^{\frac{S}{2}} \sum_{l=\frac{L}{2}}^{\frac{L}{2}} M[i + s, j + l, k, n]}{S \times L}$$

ECUACIÓN 27 PROMEDIADO POR VECINDARIO 2D

Píxel más oscuro: De la observación de texturas, es posible notar que son las intensidades más bajas las que dominan la “forma” de la textura, y definen el patrón que las compone. La homogeneización puede hacerse entonces, mediante la utilización de la sub-matriz perteneciente al pixel más oscuro de cada vecindario.

$$M[i, j] = M_o [s, l]$$

ECUACIÓN 28 PIXEL MÁS OSCURO

Donde M_o es la sub-matriz del pixel más oscuro en un vecindario $S \times L$, centrado en $[i, j]$.

Promedio más oscuro: Siguiendo el mismo principio que en el caso anterior, se utiliza en vez de la sub-matriz del pixel más oscuro, la sub-matriz en promedio más oscura para homogenizar en cada vecindario.

$$M[i,j] = M_p [s, l]$$

ECUACIÓN 29 PROMEDIO MÁS OSCURO

Donde M_p es la matriz con el promedio más oscuro en un vecindario $S \times L$, centrado en $[i,j]$.

3. Se calculan luego, los pasos de iteración mediante simulación computacional del comportamiento de la distribución f de fotones. Para esto, se sigue la Ecuación 4, que postula:

$$f_i(\vec{x} + \vec{e}_i, t + 1) = \sum_j M_{ij} f_j(\vec{x}, t), i = 0, 1, \dots, l$$

Se itera la ecuación hasta que $f(t) = f(t + 1)$, momento en el que se ha alcanzado la convergencia del sistema.

4. Se extrae información a partir de la distribución f resultante. Cualquiera de las variables macroscópicas presentadas, como el ángulo, la entropía o la velocidad, son candidatas en este paso de la detección.

5. Existe una relación entre las características de las texturas y los valores de las variables macroscópicas obtenidas en el paso anterior. Es posible realizar una segmentación por umbralado, tomando como base las magnitudes obtenidas a partir de las variables macroscópicas.

Las alternativas de configuración de algoritmo para la segmentación comprenden entonces, las variantes presentadas en el gráfico de la Figura 53. Se combinan los ocho tamaños del vecindario estudiados (ventanas de 3x3, 5x5, 7x7, 9x9, 11x11, 13x13, 15x15 y 17x17 pixeles de lado), los cuatro operadores básicos (suma, producto, modulo-resta y división) y los tres criterios de homogeneización estudiados (promediado, matriz del pixel más oscuro y matriz con promedio más bajo).

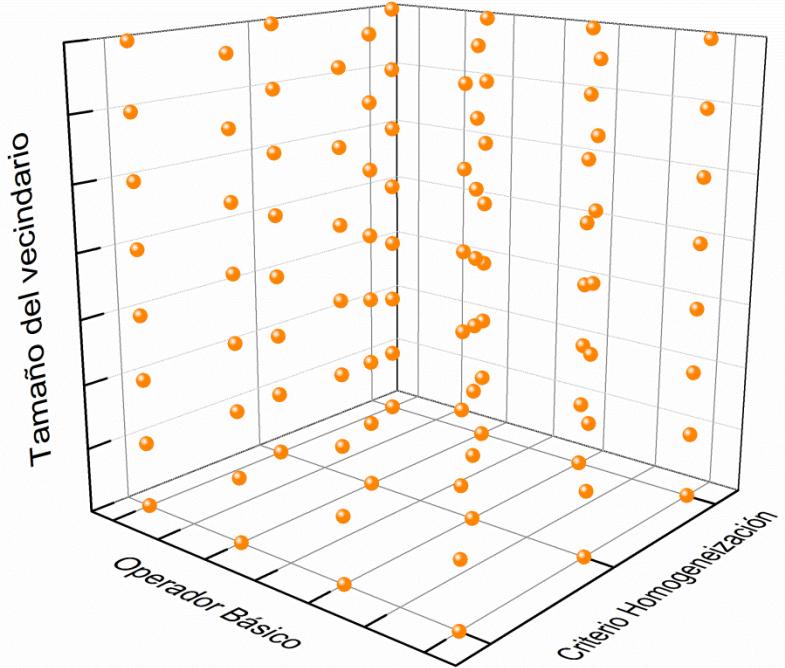


FIGURA 53 ALTERNATIVAS DE EXPERIMENTACIÓN (LOS ELEMENTOS EN CADA COLUMNAS PUEDEN COMBINARSE CON CUALQUIERA DE LAS OTRAS COLUMNAS)

La salida del algoritmo, para cualquiera de las configuraciones anteriores, es siempre la misma: una distribución f de fotones del tamaño de la imagen original sobre la que se aplica la simulación. A partir de esta salida, pueden construirse histogramas de cada una de las variables macroscópicas: densidad, entropía, velocidad, etc.

A partir de estas vistas, se puede realizar la segmentación por umbralado (que puede ser definido de diferentes maneras). Para presentar los resultados de este proceso, se utilizan:

- Gráfico de calidad de la segmentación (AOM para diferentes valores de umbral).

- Imágenes de las segmentaciones logradas (para diferentes grados de calidad alcanzados).

5.6 INDICADORES DE CALIDAD DE LA SEGMENTACIÓN

El objetivo de los experimentos descriptos es evaluar la eficiencia de cada matriz de transición para la segmentación de texturas. A fin de evaluar los resultados, a partir de una segmentación de referencia, se aplicaron métricas de comparación entre ésta y la segmentación obtenida mediante el algoritmo, las cuales han sido utilizadas en distintos trabajos como indicadores cuantitativos de calidad de segmentación (Abdel-Dayem, 2010 y del Fresno 2009). El índice en cuestión se denomina *AOM*, por *Average Overlap Metric* y viene dado por la ecuación:

$$AOM = \frac{(|V_i \cap V_a|)}{\frac{1}{2}(|V_i| + |V_a|)}$$

ECUACIÓN 30 INDICADOR AOM DE CALIDAD DE SEGMENTACIÓN

donde V_i es el conjunto de referencia de píxeles clasificados como imagen con una cierta textura y V_a es el conjunto de píxeles detectado por el algoritmo. *AOM* se define como la proporción de los píxeles que han sido asignados por ambas segmentaciones, relativo al número de píxeles de una y otra segmentación. Esta métrica alcanza el valor 1 cuando ambos resultados coinciden y es cercana a 0 si no comparten píxeles clasificados en forma similar.

Si bien el índice *AOM* es útil a la hora de comparar dos imágenes segmentadas (una por el algoritmo y otra de referencia), en los experimentos también se ha tenido la nece-

sidad de realizar comparaciones de calidad sobre las nubes de puntos formadas por cada variable macroscópica.

Dada una segmentación de referencia, puede saberse en una nube de puntos de datos provenientes de las variables macroscópicas, a qué segmento de la imagen corresponde cada uno. Es así que resulta deseable saber qué tan bien el algoritmo ha hecho el trabajo de separar dichos puntos en nubes fácilmente distinguibles por una segmentación por umbralado.

A partir de dos nubes de puntos en un plano de parámetros p y q (por ejemplo, ángulo y densidad), cada nube corresponde a cada textura diferente que se desea estén bien diferenciadas. El indicador de la calidad de segmentación viene dado por:

$$I_s = \left(\frac{p_1 - p_2}{\sigma_p} \right)^2 + \left(\frac{q_1 - q_2}{\sigma_q} \right)^2$$

ECUACIÓN 31 INDICADOR DE CALIDAD A PARTIR DE NUBES DE PUNTOS EN UN PLANO DE PARÁMETROS p Y q

donde p_i es el promedio de p sobre la nube i , q_i el promedio de q sobre la nube i y σ la desviación estándar de p sobre el total de los puntos (ambas nubes). El índice resultante I_s es una medida de la calidad de la segmentación.

6. RESULTADOS NUMÉRICOS

A continuación, se desarrollarán una serie de pruebas, de creciente complejidad, con las que ha sido posible comprender el alcance y efectividad del método planteado, así como también, las limitaciones existentes. Los casos a estudiar son los siguientes:

- Texturas Sintéticas
 - Textura Homogénea: Imagen compuesta por un mismo patrón simple que se repite a lo largo de toda la imagen.
 - Textura Compuesta: Imágenes compuestas por más de una textura, con la consecuente aparición de bordes que pueden ser simples o complejos, dependiendo del caso.
 - Textura Procedural: Imágenes con texturas procedurales, que se asemejen a las capturadas en fotografías de materiales y objetos reales.
- Texturas mixtas
 - Imágenes compuestas por una textura sintética y otra real.
- Texturas Reales
 - Fotografías de texturas: Imágenes reales, capturadas con cámaras fotográficas convencionales, con presencia de patrones repetitivos.

Se hicieron experimentos con todas las configuraciones diferentes, y se consideró, en cada caso, aquellos parámetros que maximizaron el índice de segmentación propuesto en la sección anterior. Los tamaños del vecindario se eligieron conforme a la textura que se esté utilizando en cada momento para el experimento y, en el caso de las texturas naturales donde no se conoce de antemano un valor óptimo de vecindario, se hicieron pruebas hasta encontrar aquel que maximiza la calidad de los resultados. Por otra parte, el umbralado se realiza a partir de los datos provenientes de la o las variables que mejor se comportaron al momento de realizar la distinción de texturas.

6.1 CASO 1: TEXTURA HOMOGÉNEA

Al principio de la Sección 5 se consideró que, una textura homogénea debía estar representada por un solo punto. Esto motivó la realización de un promediado por vecindario que, homogeneizó la matriz de características, pero que volvía 0 las velocidades luego de la simulación. La Figura 54 muestra la textura en escala de grises utilizada en este experimento.

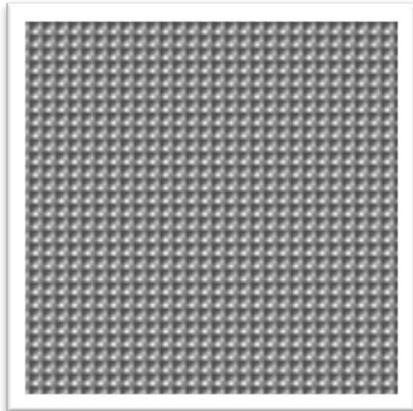


FIGURA 54 TEXTURA HOMOGÉNEA EN ESCALA DE GRISES

Al realizar los experimentos con la solución propuesta, fue posible observar una total mejoría de los resultados del algoritmo.

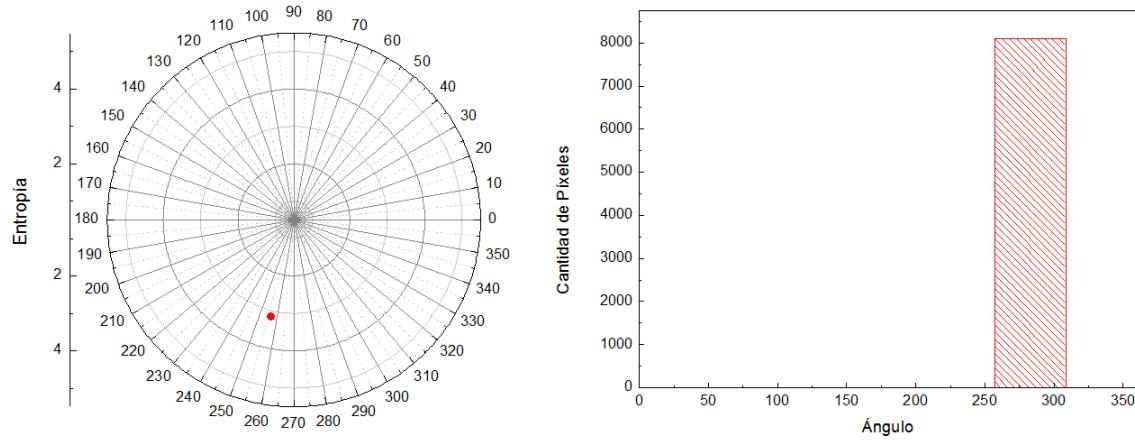


FIGURA 55 ÁNGULO Y ENTROPÍA (IZQUIERDA), HISTOGRAMA DE ÁNGULOS (DERECHA) PARA UNA TEXTURA HOMOGÉNEA

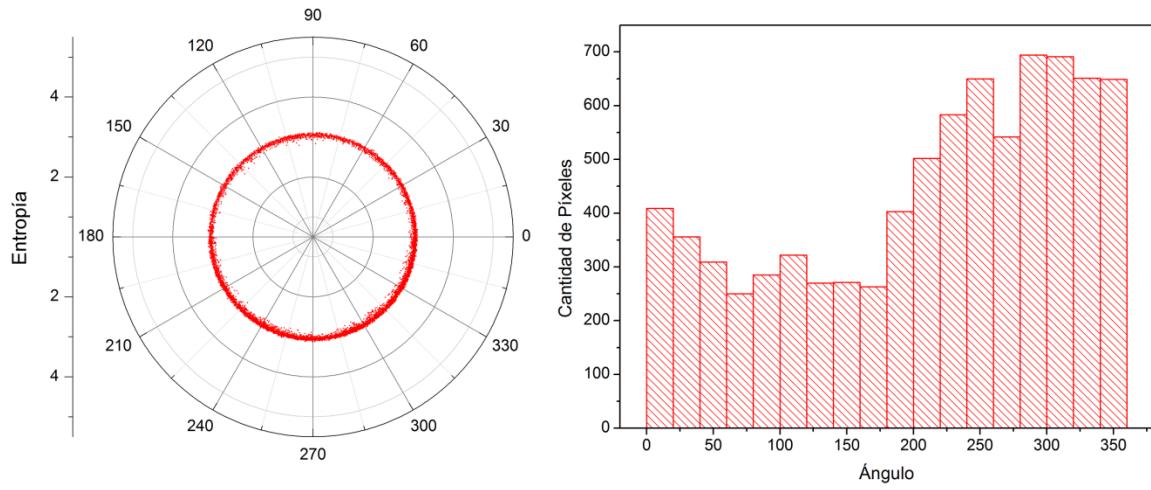


FIGURA 56 ÁNGULO Y ENTROPIA (IZQUIERDA), HISTOGRAMA DE ÁNGULOS (DERECHA) PARA UNA TEXTURA HOMOGÉNEA CON RUIDO DEL 1%

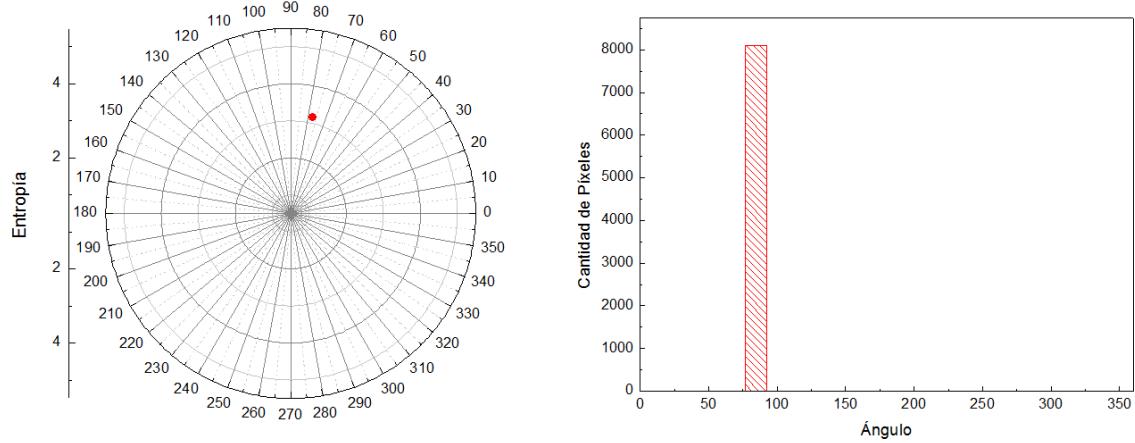


FIGURA 57 ÁNGULO Y ENTROPIA (IZQUIERDA), HISTOGRAMA DE ÁNGULOS (DERECHA) PARA UNA TEXTURA HOMOGÉNEA ROTADA

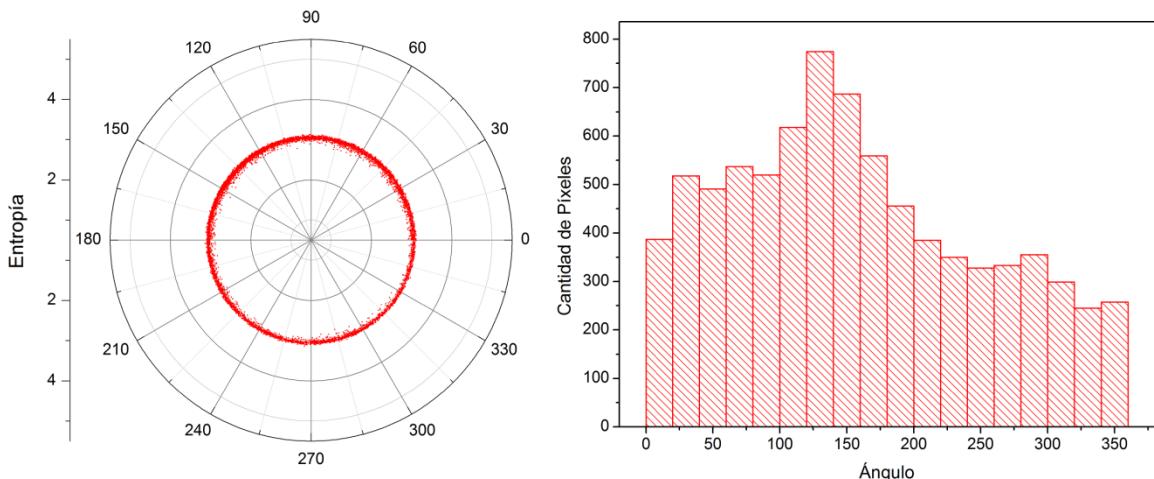


FIGURA 58 ÁNGULO Y ENTROPIA (IZQUIERDA), HISTOGRAMA DE ÁNGULOS (DERECHA) PARA UNA TEXTURA HOMOGÉNEA ROTADA CON RUIDO DEL 1%

La Figura 55 muestra el ángulo y la entropía asociados a la textura homogénea, y la Figura 57, a la misma textura rotada 180 grados. Por otro lado, en los casos sin presencia de ruido, los histogramas no muestran una dispersión de los valores por regiones, sino una sola concentración en un solo valor de ángulo.

Cuando se introduce ruido en ambas texturas (Figura 56 y Figura 58), la gráfica del ángulo y la entropía deja de ofrecer un solo punto, para mostrar todo un espectro de ángulos, que barren por completo el espacio angular. En los histogramas puede verse más claramente que donde antes se tenía una sola gran concentración, ahora hay una distribución no tan concentrada en un punto, y que disminuye conforme nos alejamos de él.

6.2 CASO 2: TEXTURA COMPUESTA CON BORDE SIMPLE

Al realizar pruebas sobre imágenes más complejas (Figura 59), compuestas por las dos texturas presentadas anteriormente, los resultados son igual de efectivos.

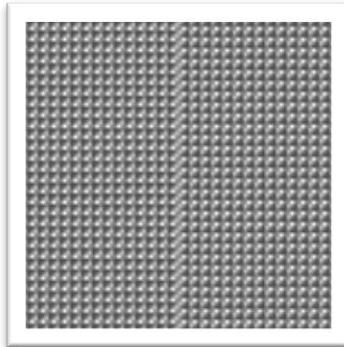


FIGURA 59 IMAGEN COMPUESTA, CON DOS TEXTURAS EN ESCALA DE GRIS

La Figura 60 muestra cómo los ángulos separan los píxeles en dos grupos claramente definidos, detectando que se trata de las dos texturas diferentes de la Figura 59. Al realizar una segmentación por umbralado (como la de la Sección 4.3) utilizando un ángulo límite, puede calcularse el índice *AOM* para dicha segmentación. Esta segmentación se realiza a partir del campo angular extraído luego de la ejecución del algoritmo y eligiendo un ángulo de inicio (α) y una apertura (β). Todos aquellos valores del campo angular que entran en ese rango ($\alpha, \alpha + \beta$), serán considerados parte de una textura, y los que no, como parte de la otra textura que compone la imagen. A medida que el ángulo de inicio es desplazado, nuevos elementos entran y salen del conjunto. Este corrimiento se realiza sistemáticamente, hasta cubrir todo el espacio y encontrar un valor óptimo del índice *AOM*, que implica haber separado en dos conjuntos distinguibles las texturas de la imagen.

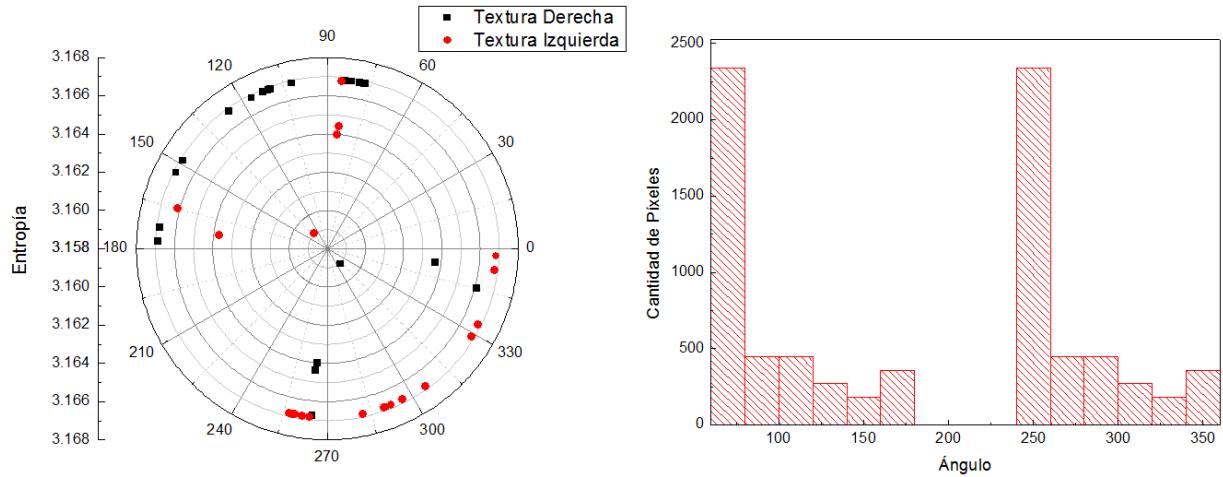


FIGURA 60 ÁNGULO Y ENTROPÍA (IZQUIERDA), HISTOGRAMA DE ÁNGULOS (DERECHA) PARA UNA TEXTURA COMPUSTA

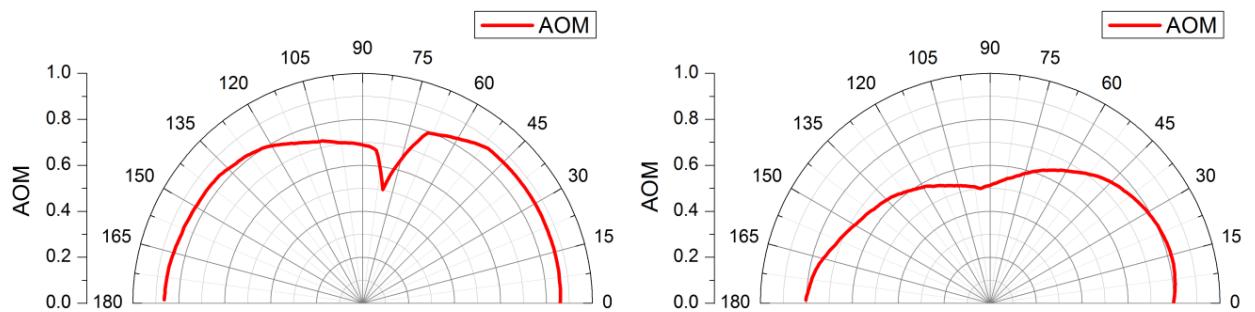


FIGURA 61 VALORES DE AOM PARA DIFERENTES ÁNGULOS DE UMBRAL (INICIO EN 0 Y VENTANA DE 180°) PARA LA IMAGEN SIN RUIDO (IZQUIERDA), Y CON RUIDO DEL 1% (DERECHA)

Para el cálculo del índice AOM en este caso, se consideró una apertura de 180° con inicio en 0, que fue desplazado hasta llegar a 180° . En la Figura 61 puede observarse como a medida que se corre el umbral de segmentación, su calidad disminuye, hasta que finalmente rebota, debido a la simetría axial existente en la distribución de ángulos (los valores para cada una de las texturas se encuentran enfrentados en el plano polar).

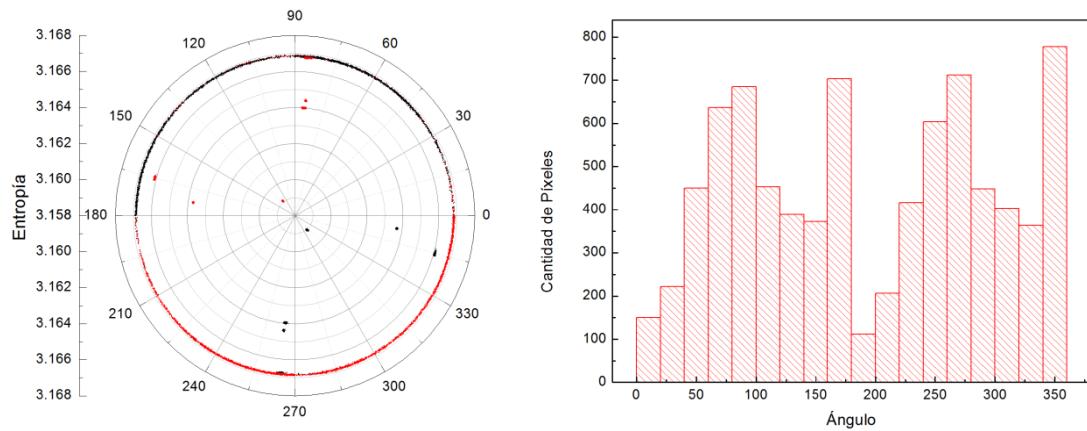


FIGURA 62 ÁNGULO Y ENTROPÍA (IZQUIERDA), HISTOGRAMA DE ÁNGULOS (DERECHA) PARA UNA TEXTURA COMPUESTA CON RUIDO = 1%

Para este caso, el valor óptimo de *AOM* para evaluar la calidad de la segmentación es 0.827489. Este valor empeora a medida que se aumenta la cantidad de ruido presente en la imagen, volviéndose 0.813379 para un ruido de 1%, y 0.49064% para ruido del 10%. En la Figura 62 puede observarse como la introducción de ruido en la imagen de prueba hace que los ángulos se distribuyan de manera más uniforme en el plano polar (a diferencia de lo que se ve en la Figura 60), complicando la tarea de segmentación y disminuyendo el valor óptimo obtenido mediante el índice *AOM*.

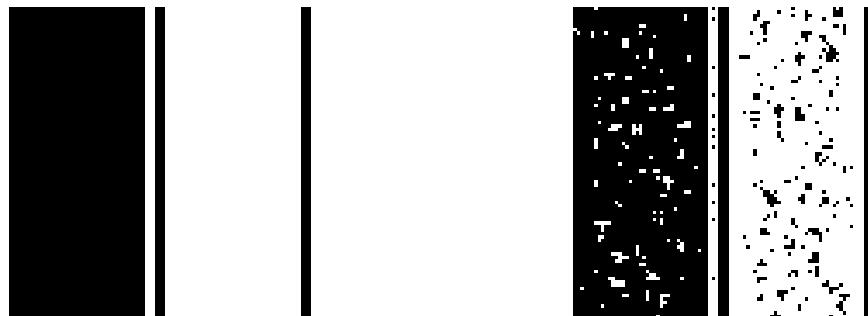


FIGURA 63 SEGMENTACIÓN ÓPTIMA POR UMBRALADO PARA LA IMAGEN COMPUSTA SIN RUIDO (IZQUIERDA), Y CON RUIDO DEL 1% (DERECHA)

En la Figura 63 es posible observar las segmentaciones (para los valores máximos de AOM logrados) para la textura con y sin ruido. Las imágenes binarias muestran, además de los “artefactos” introducidos por el ruido, el comportamiento del algoritmo ante la presencia de bordes. Estas “líneas” que aparecen antes de un cambio en la textura están relacionadas con el vecindario utilizado y la forma en que se calcula la matriz del material.

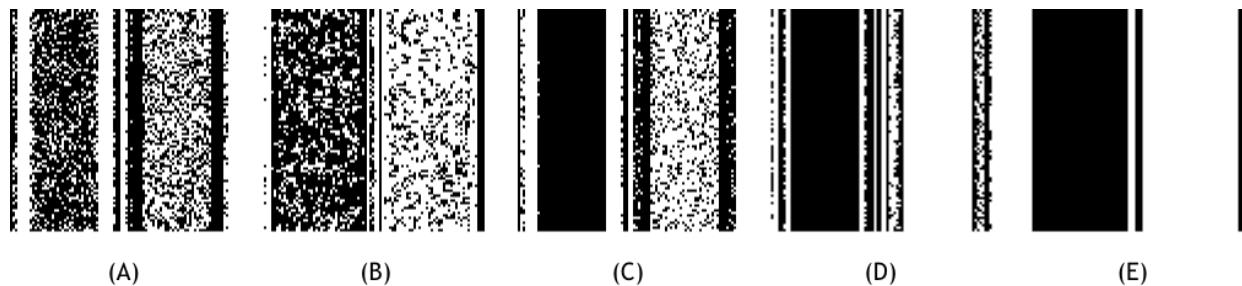


FIGURA 64 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL. EL ÍNDICE AOM ES DE 0.5 (A), 0.6 (B), 0.7 (C), 0.8 (D) Y 0.82 (E)

Al observar las segmentaciones para diferentes valores de umbral (Figura 64 y Figura 65), es posible distinguir que, a medida que se eligen mejores valores de umbral, hay un aumento del índice *AOM* y consecuentemente, una mejora en la calidad de la segmentación. Si bien la calidad de la mejor segmentación para la figura sin ruido es muy buena (Figura 64 E), no es posible alcanzar un valor óptimo de *AOM* (cercano a 1), debido al comportamiento del algoritmo en los bordes entre ambas texturas.

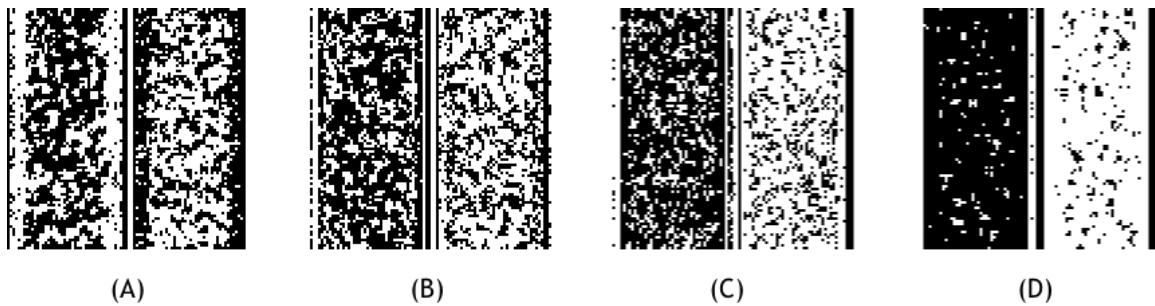


FIGURA 65 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL (FIGURA CON RUIDO). EL ÍNDICE *AOM* ES DE 0.5 (A), 0.6 (B), 0.7 (C) Y 0.8 (D)

6.3 CASO 3: TEXTURA COMPUESTA CON BORDE COMPLEJO

Hasta aquí, las imágenes utilizadas poseían bordes simples entre texturas. Para testear la respuesta del algoritmo frente a bordes más complejos, se utilizó la imagen de la Figura 66, de igual composición que la de la Figura 59, pero con borde en zigzag.

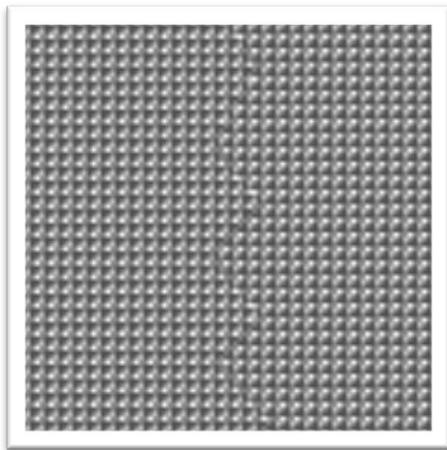


FIGURA 66 IMAGEN CON DOS TEXTURAS Y UN BORDE COMPLEJO

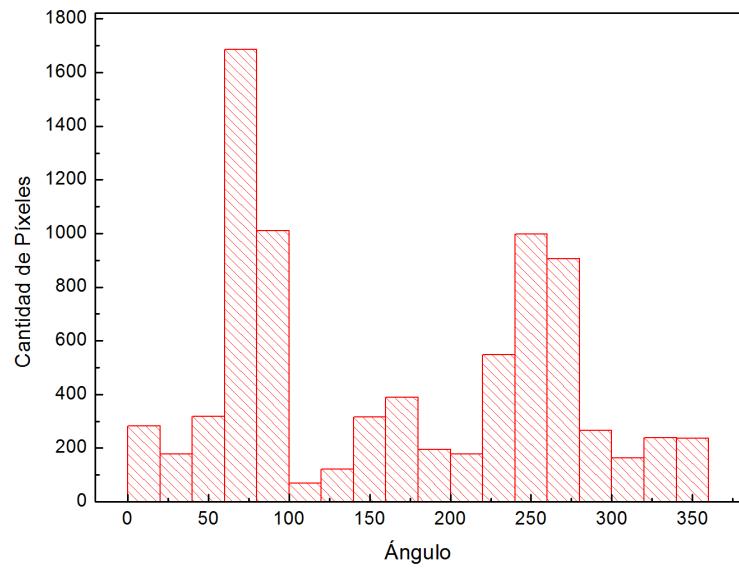


FIGURA 67 HISTOGRAMA DE ÁNGULOS PARA LA IMAGEN COMPUESTA CON BORDE COMPLEJO

Los resultados (Figura 67) fueron similares a los obtenidos con el borde simple. Se realizó la segmentación por umbralado y el valor de *AOM*, para evaluar la calidad de la segmentación, fue de 0.894325. Este valor empeora a medida que se aumenta la cantidad de ruido presente en la imagen, volviéndose 0.813379 para un ruido de 1%.

Al igual que en el ejemplo anterior, en la Figura 68 puede observarse como a medida que se corre el umbral de segmentación, su calidad disminuye, hasta que finalmente mejora, debido a que también existe una simetría axial en la distribución de ángulos. Se utilizó una apertura de 180°, que fue desplazada desde el 0° hasta los 180°.

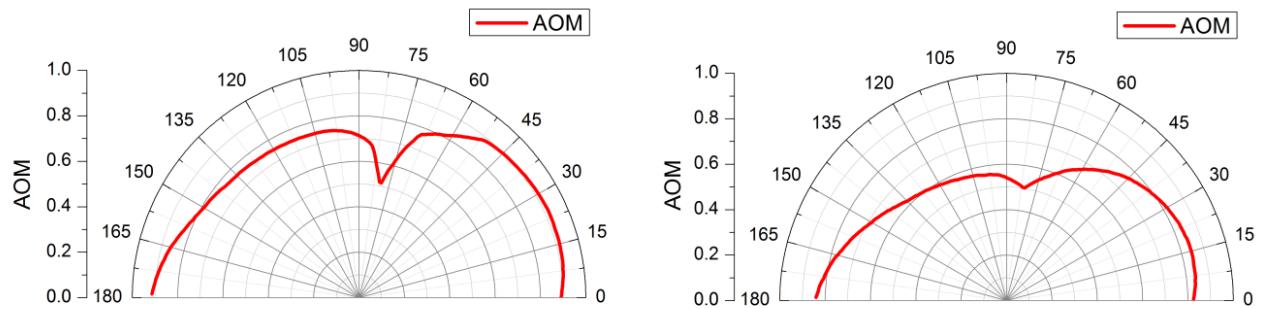


FIGURA 68 VALORES DE AOM PARA DIFERENTES ÁNGULOS DE UMBRAL PARA LA IMAGEN SIN RUIDO (IZQUIERDA), Y CON RUIDO DEL 1% (DERECHA)

Las segmentaciones para los valores máximos de *AOM* pueden observarse en la Figura 69. Por un lado, la imagen sin ruido solo presenta algunas perturbaciones en la calidad de la detección debido al borde irregular empleado en este caso. En cambio en la imagen ruidosa, si bien pueden distinguirse claramente dos regiones bien definidas, la interferencia de pequeños errores en la detección empeora la calidad de la segmentación.



FIGURA 69 SEGMENTACIÓN ÓPTIMA POR UMBRALADO PARA LA IMAGEN COMPUSTA CON BORDE COMPLEJO SIN RUIDO (IZQUIERDA), Y CON RUIDO DEL 1% (DERECHA)

Por otro lado, si observamos la evolución de las segmentaciones para diferentes valores de AOM (Figura 70), podemos apreciar cómo las regiones segmentadas se vuelven cada vez más claras y uniformes a medida que el valor de dicho índice se incrementa.

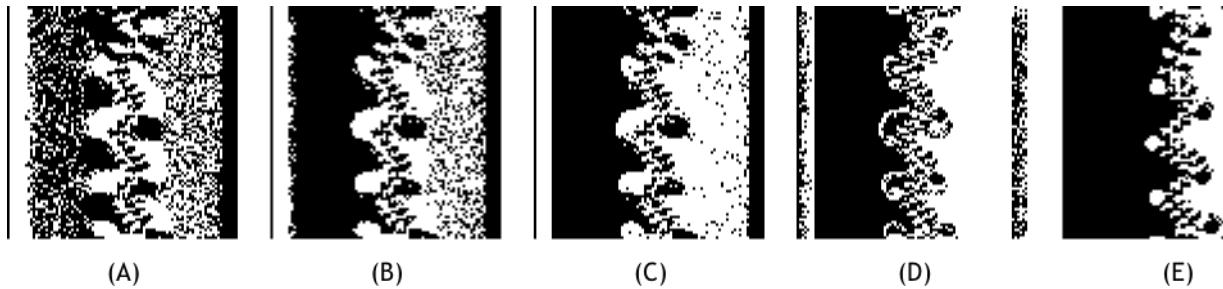


FIGURA 70 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL. EL ÍNDICE AOM ES DE 0.5 (A), 0.6 (B), 0.7 (c), 0.8 (D), 0.89 (E)

Es importante notar que, a pesar de que AOM indique baja concordancia entre la región segmentada y la región a segmentar, para valores por encima del 0.6 ya es posible

distinguir entre las texturas que componen la imagen (Figura 70 B). Esto significa que es posible detectar y segmentar regiones de textura inclusive con un umbral no tan bueno como el que se logra en el caso óptimo presentado en la Figura 69.

Ante la presencia de ruido, es posible distinguir entre las dos texturas que componen la imagen, a partir umbrales que generen valores de *AOM* superiores a 0.7 (Figura 71).

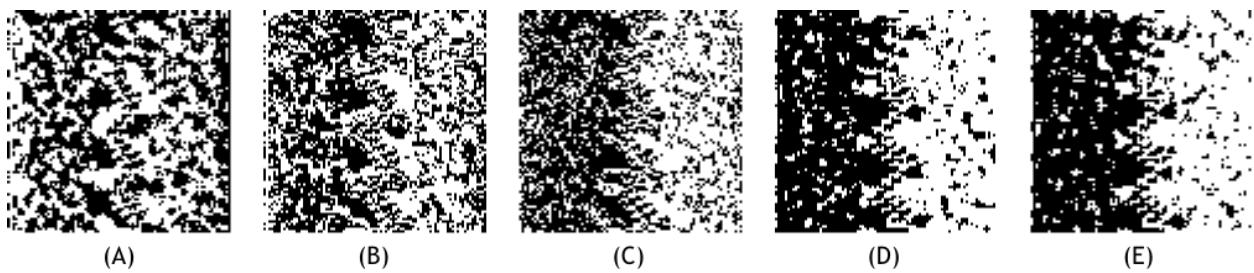


FIGURA 71 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL. TEXTURA CON RUIDO DEL 1% EL ÍNDICE AOM ES DE 0.5 (A), 0.6 (B), 0.7 (C), 0.8 (D), 0.81 (E)

6.4 CASO 4: TEXTURAS PROCEDURALES

6.4.1 TEXTURA HOMOGÉNEA

Es también deseable, estudiar el comportamiento del método cuando se lo expone a texturas parecidas a las encontradas en la realidad. Una de las aplicaciones industriales más importantes del procesamiento digital de imágenes, es la detección de fallas en la fabricación de géneros por parte de la industria textil. Encontrar diferencias en los patro-

nes homogéneos que las componen, puede ser interesante para determinar la calidad de las telas.

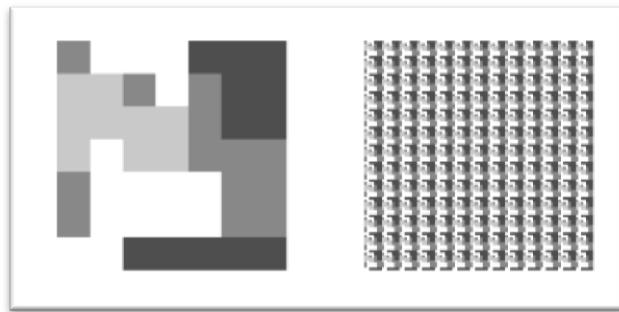


FIGURA 72 PATRÓN DE BASE DE 7x7 (IZQUIERDA) Y TEXTURA GENERADA A PARTIR DE ELLA (DERECHA)

Para verificar el comportamiento del algoritmo en este tipo de imágenes, se ha diseñado un experimento a partir de la textura de la Figura 72. Trabajamos ahora con un vecindario de 7x7 para generar la textura, motivo por el cual el vecindario promediado se realiza entre 49 celdas (y no entre 9). Este cambio se realizó para tener más libertad a la hora de generar texturas parecidas a las encontradas en imágenes digitales reales. Así, es posible lograr texturas similares a las que se encuentran en las telas de producción industrial.

Al aplicar el algoritmo sobre estas imágenes, el resultado es el esperado. La Figura 73 muestra la concentración de ángulos en un solo punto del gráfico, tal como fue visto en otras texturas homogéneas.

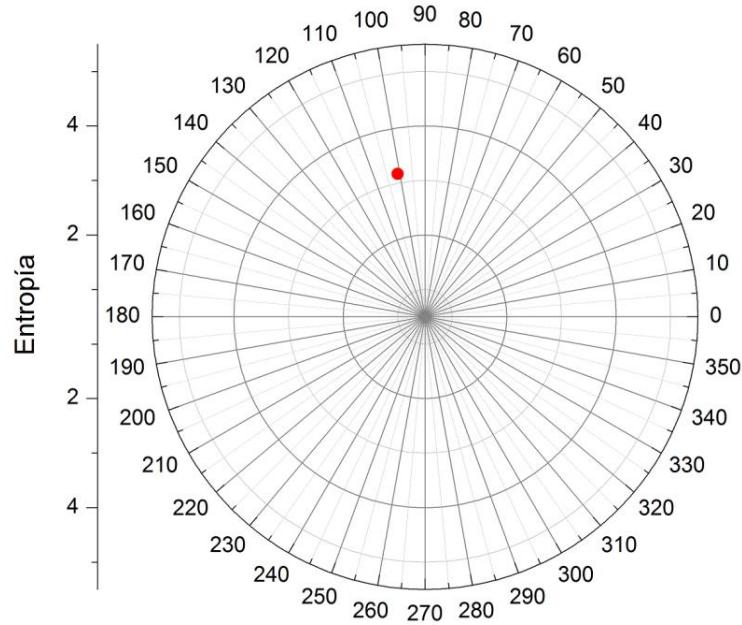


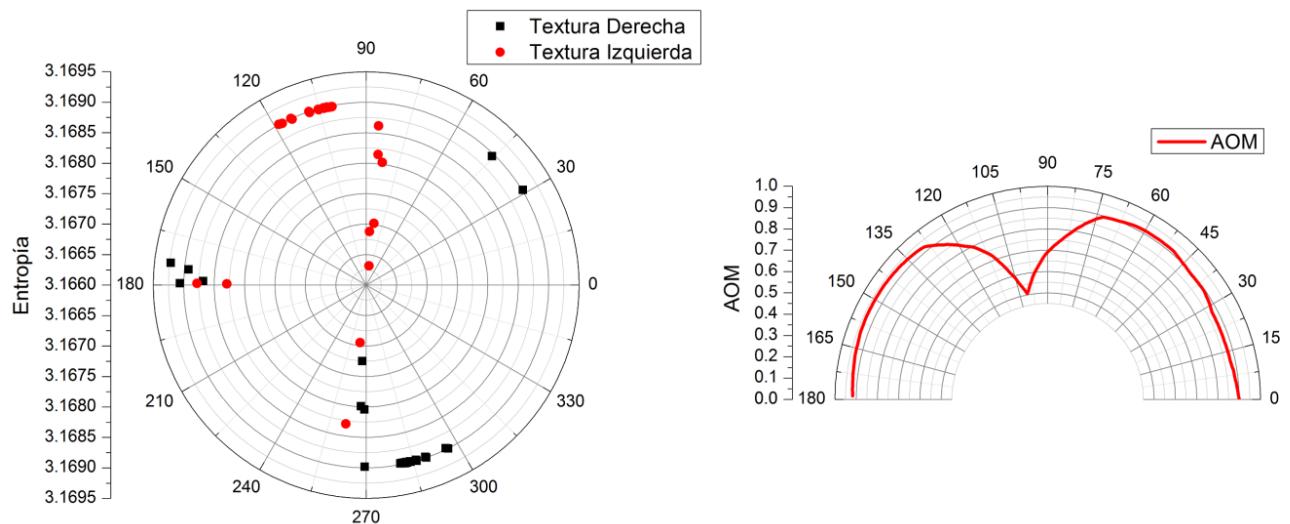
FIGURA 73 GRÁFICA DEL ÁNGULO Y LA ENTROPIA, LUEGO DE APLICAR EL ALGORITMO DE DETECCIÓN SOBRE UNA TEXTURA DE 7x7

6.4.2 TEXTURA COMPUUESTA SIMPLE

A partir de la textura de la sección anterior, se generó una nueva imagen (ver Figura 74), que está compuesta por dos rotaciones de la textura anterior. Los resultados también fueron satisfactorios al aplicar el algoritmo de detección y segmentación. EL índice *AOM* alcanzó un pico de 0.91 para la imagen en cuestión (Figura 75).



FIGURA 74 MÁSCARA (IZQUIERDA) Y TEXTURA COMPUESTA CON PATRÓN BASE DE 7x7 (DERECHA)



**FIGURA 75 GRÁFICA DE ÁNGULO Y ENTROPIA (IZQUIERDA) Y ÍNDICE AOM PARA DIFERENTES ÁNGULOS DE UMBRAL (DERECHA)
PARA LA SEGMENTACIÓN SIN PRESENCIA DE RUIDO.**

La mejor segmentación lograda (Figura 76) es similar a la obtenida con las texturas de la Sección 6.2, tanto en el caso sin presencia de ruido, como en el que se introdujo ruido del 1%.

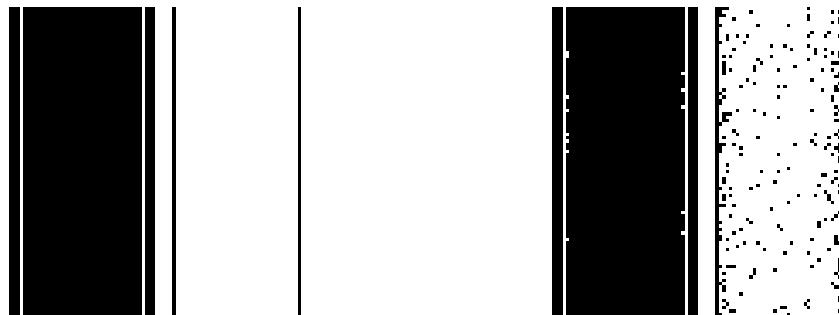


FIGURA 76 SEGMENTACIÓN ÓPTIMA POR UMBRALADO PARA LA IMAGEN COMPUESTA SI RUIDO (IZQUIERDA), Y CON RUIDO DEL 1% (DERECHA)

Por otro lado, al igual que en los casos estudiados anteriormente, a medida que se utilizan umbrales con mejores valores de AOM , la segmentación se vuelve más certera (Figura 77 y Figura 78). El óptimo absoluto no es alcanzado, debido a que los bordes no logran estar bien definidos, consecuencia directa de la utilización del promediado por vecindario, que hace que las matrices generadas en los bordes entre texturas sean una mezcla de cada una de las texturas que lindan con dicho borde.

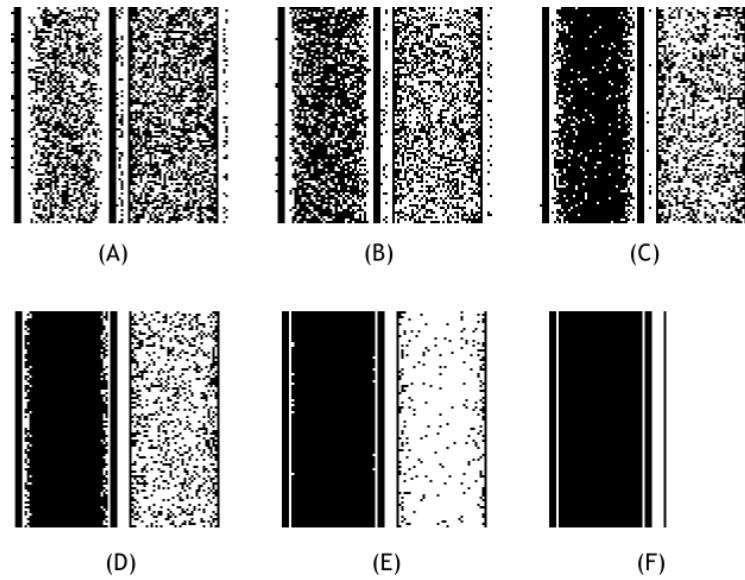


FIGURA 77 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL (IMAGEN SIN RUIDO). EL ÍNDICE AOM ES DE 0.5 (A), 0.6 (B), 0.7 (C), 0.8 (D), 0.9 (E), 0.91 (F)

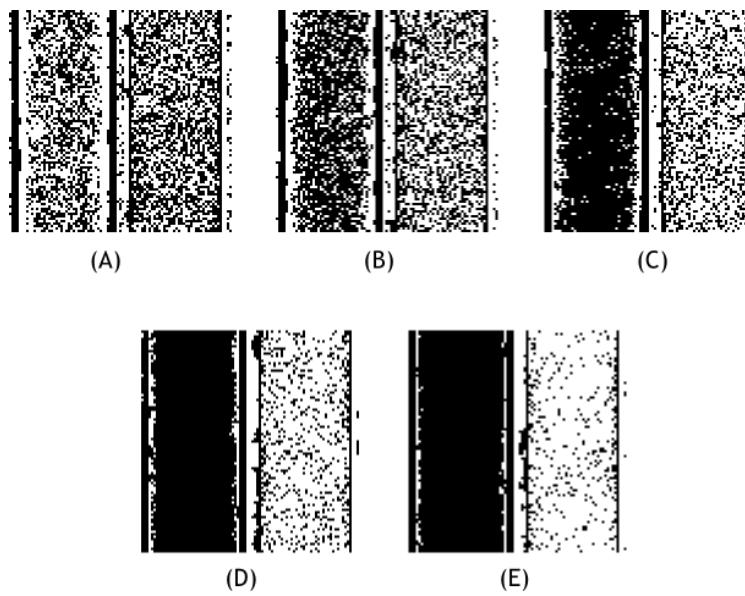


FIGURA 78 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL (RUIDO 1%). EL ÍNDICE AOM ES DE 0.5 (A), 0.6 (B), 0.7 (C), 0.8 (D), 0.81 (E)

6.4.3 TEXTURA COMPUESTA COMPLEJA

Adicionalmente, nos interesa probar con texturas basadas en imágenes reales de géneros. La Figura 79 muestra una imagen del conocido patrón “Pata de Gallo”, muy utilizado en la industria textil.

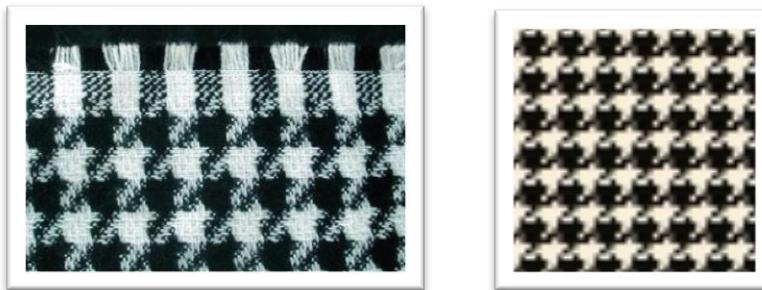


FIGURA 79 PATRÓN DE TELA "PATA DE GALLO" (IZQUIERDA) Y SU EQUIVALENTE PROCEDURAL (DERECHA)

En este caso, se realizaron pruebas directamente sobre imágenes ruidosas, de manera que simulen adecuadamente las condiciones reales del entorno de captura de la imagen.

Por un lado, se tomó la textura con un 1% de ruido (Figura 79, derecha), y por otro, a la misma textura se le introdujo una región irregular con 20% de ruido para simular el gastado de la tela (Figura 81). Si bien las gráficas de ángulo y entropía no parecen separar a simple vista ambas regiones (ver Figura 80), el gráfico del índice AOM (Figura 82), muestra que esta distinción parece ser realizada por el algoritmo, con valores por encima del 75% de la región segmentada correctamente.

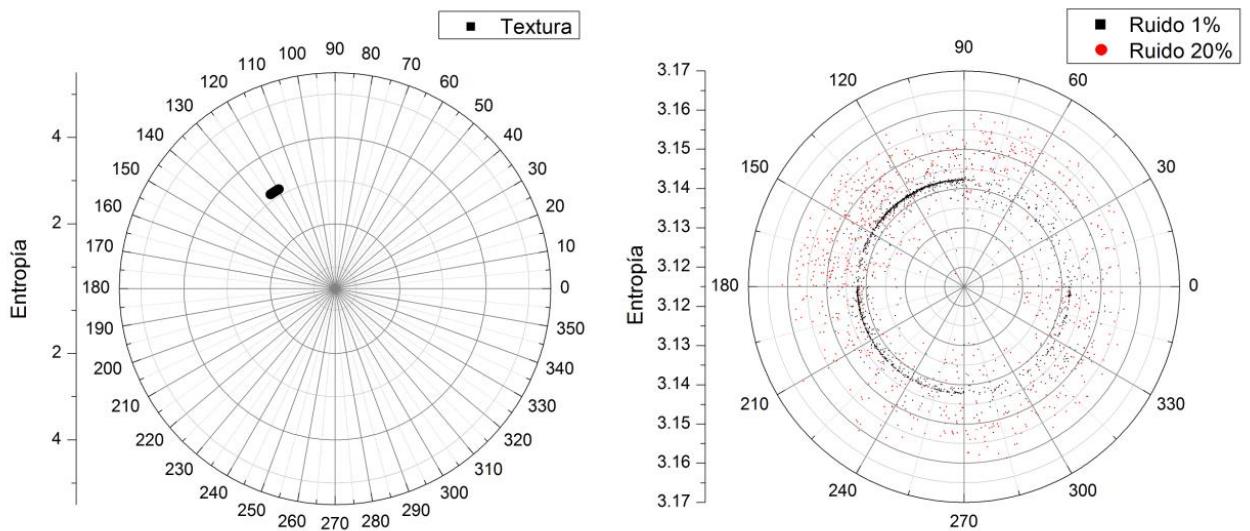


FIGURA 80 ÁNGULO Y ENTROPÍA PARA TEXTURA "PATA DE GALLO" CON 1% RUIDO (IZQUIERDA) Y CON REGIÓN CON 20% DE RUIDO (DERECHA)

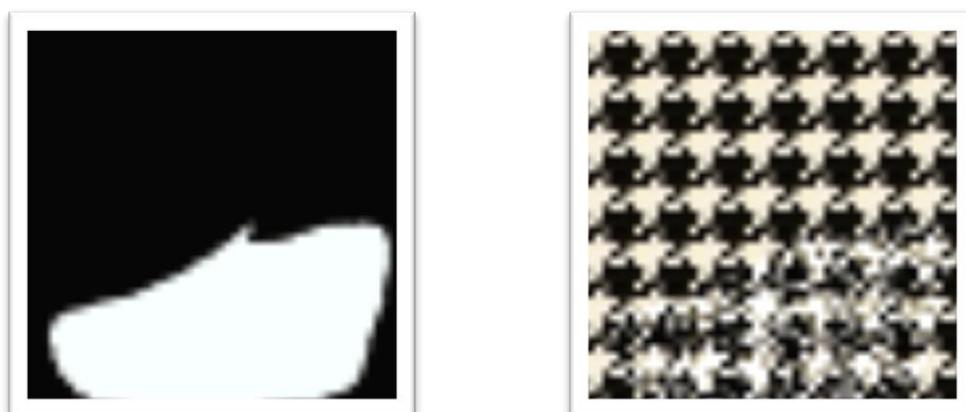


FIGURA 81 MASCARA PARA SIMULAR EL GASTADO (IZQUIERDA) Y PATRÓN DE TELA "PATA DE GALLO" CON 20% DE RUIDO EN ZONA IRREGULAR (DERECHA)

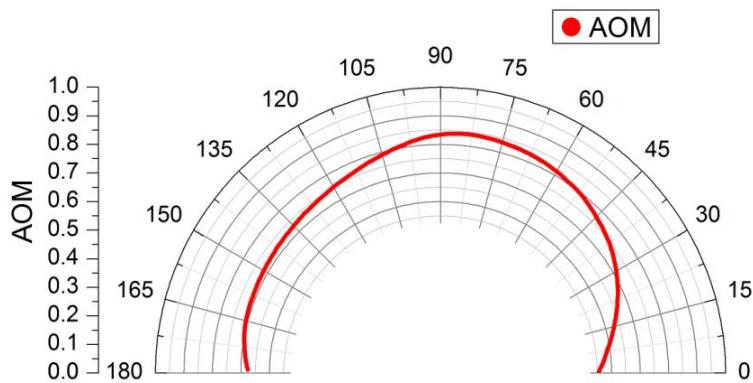


FIGURA 82 ÍNDICE AOM PARA LA TEXTURA “PATA DE GALLO” CON REGIONES CON RUIDO

Sin embargo, al observar las segmentaciones logradas (Figura 83) se ve que el algoritmo ha fallado al realizar la detección. Si bien hay una clara mejoría a medida que el índice AOM se incrementa, 0.75 resulta ser un ajuste demasiado bajo para la figura. No obstante, la mitad superior, carente de regiones ruidosas, se ve mas homogénea que la mitad inferior que contiene la zona conflictiva.



FIGURA 83 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL. EL ÍNDICE AOM ES DE 0.5 (A), 0.6 (B), 0.7 (c), 0.75 (D)

6.5 CASO 5: FOTOGRAFÍAS DE TEXTURAS

También se utilizaron fotografías de materiales reales (en este caso, telas), para verificar el comportamiento del algoritmo en estos casos.

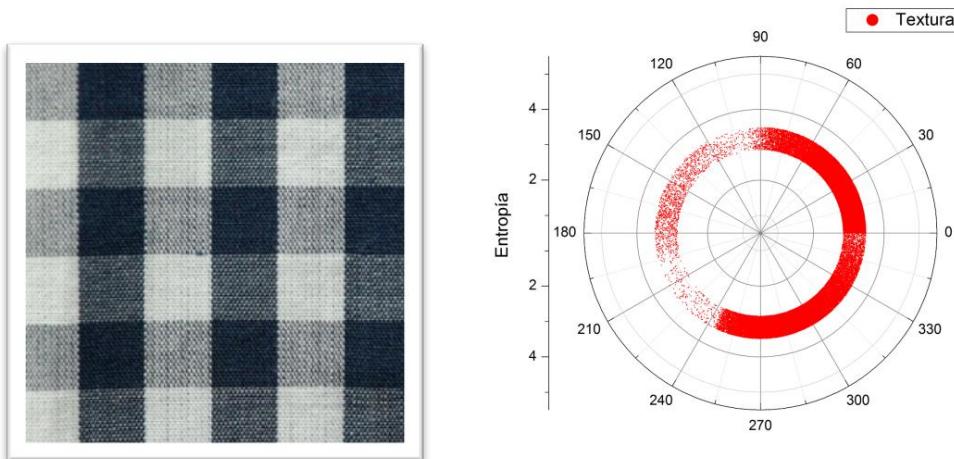


FIGURA 84 FOTOGRAFÍA DE UNA TELA DE TEXTURA HOMOGÉNEA (IZQUIERDA), ÁNGULO Y DENSIDAD RESULTANTES DE LA EJECUCIÓN DEL ALGORITMO (DERECHA)

En primer lugar, se hicieron los experimentos con la textura de la Figura 84 (izquierda), una tela de textura homogénea. El tamaño del vecindario fue adaptado para que coincidiera con el de la figura. Puede observarse que la distribución de ángulos se encuentra concentrada en un área que caracteriza a la textura en estudio. Debido a la presencia de ruido y de otras irregularidades en la fotografía utilizada, esta área ocupa

gran parte del conjunto de valores posibles, alejándose de la situación ideal (concentración en un solo punto).

Para verificar si era posible realizar una segmentación de una sección irregular en este tipo de texturas, se realizaron nuevas ejecuciones del algoritmo, utilizando la textura de la Figura 85 (Izquierda).

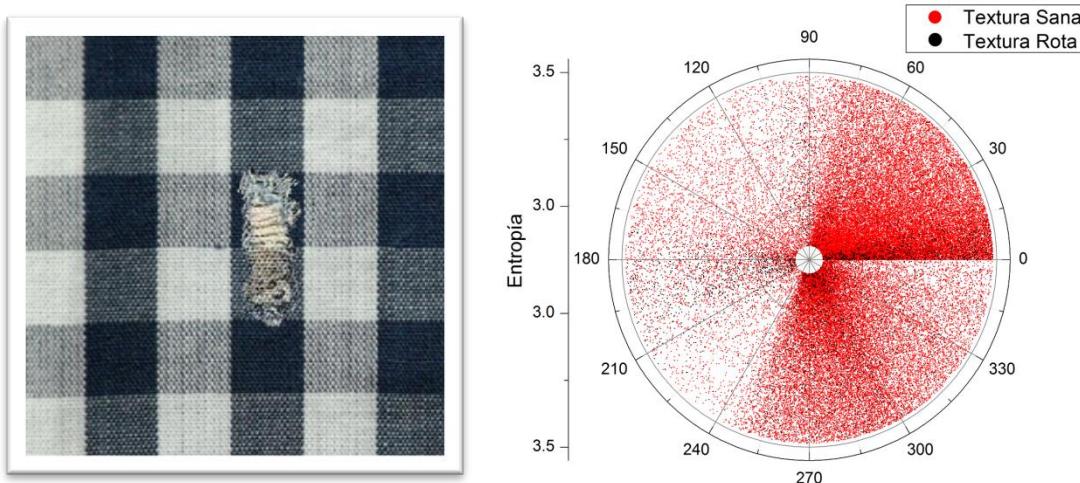


FIGURA 85 FOTOGRAFÍA DE UNA TELA DE TEXTURA COMPUSTA (IZQUIERDA), ÁNGULO Y DENSIDAD RESULTANTES DE LA EJECUCIÓN DEL ALGORITMO (DERECHA)

EL índice *AOM* calculado para diferentes valores de umbral (Figura 86) mostró que el algoritmo era capaz de lograr una segmentación de la sección irregular con una precisión por encima del 75 %. Esta segmentación puede constatarse a través de los gráficos de la Figura 87.

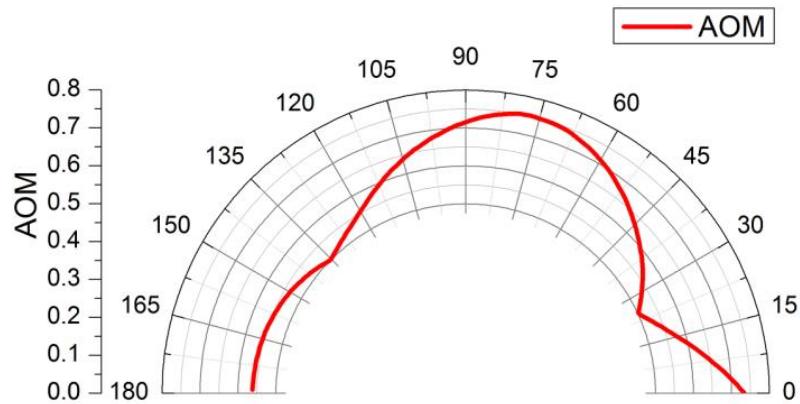


FIGURA 86 ÍNDICE AOM PARA DIFERENTES ÁNGULOS DE UMBRAL PARA LA TEXTURA DE LA FIGURA 72

En este caso, la segmentación lograda es de mejor calidad que la de la sección anterior aunque, por tratarse de imágenes mas complejas, aún no se logra una calidad óptima. Pueden observarse los efectos del ruido (puntos aislados) y de irregularidades en el tamaño del vecindario elegido (rayas aisladas).

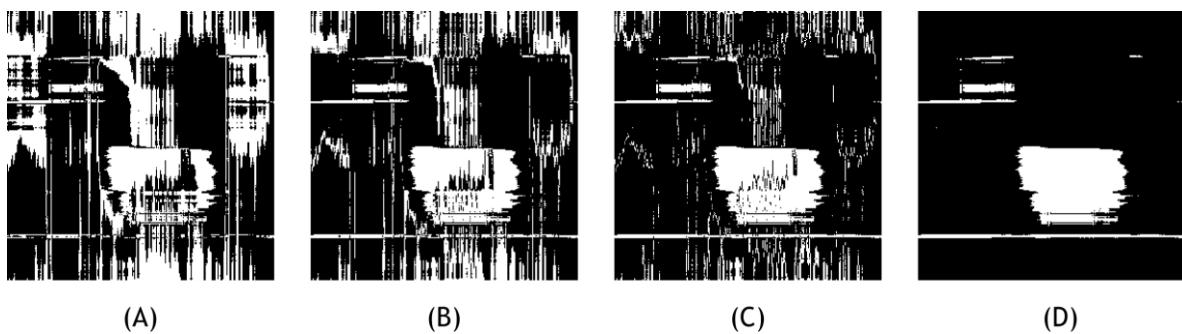


FIGURA 87 SEGMENTACIÓN PARA DIFERENTES ÁNGULOS DE UMBRAL. EL ÍNDICE AOM ES DE 0.5 (A), 0.6 (B), 0.7 (C), 0.75 (D)

6.6 CASO 6: IMAGEN CON TEXTURAS SINTÉTICO-NATURALES

Con el objeto de testear el algoritmo sobre imágenes digitales que incluyeran elementos reales tomados por cámaras fotográficas, se hicieron pruebas sobre una imagen digital (en escala de grises) construida con texturas sintéticas y follaje natural, buscando discriminar un tipo del otro (ver Figura 88). Se simularon todas las posibles combinaciones entre operadores (descriptos en las ecuaciones de operadores), vecindarios (desde 3x3 hasta 17x17 pixeles de dimensión) y variantes para seleccionar M , la matriz de probabilidad de transición entre direcciones más representativa del vecindario (Ecuación 27 a Ecuación 29).

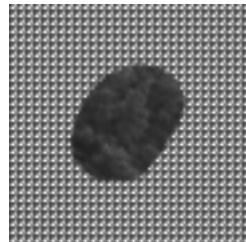


FIGURA 88 IMAGEN DE ESTUDIO QUE INTEGRA TEXTURAS SINTÉTICAS Y NATURALES.

Una vez realizada la simulación sobre la imagen de prueba, se utilizó el índice de calidad explicitado en la Sección 5.6. Se evaluaron todos los pares posibles entre los parámetros: densidad, velocidad, módulo de la velocidad, entropía, ángulo y energía interna.

En coherencia con los gráficos percibidos visualmente, los valores significativos de I_s se hallaron en los pares en donde participaba la entropía; en particular, asociada al

estudio del operador suma (Ecuación 23) junto a la matriz estocástica de probabilidad de transición entre direcciones homogeneizada (Ecuación 27). También se observó que, para esta configuración, la entropía separaba ambas texturas independientemente del tamaño del vecindario seleccionado (Figura 89).

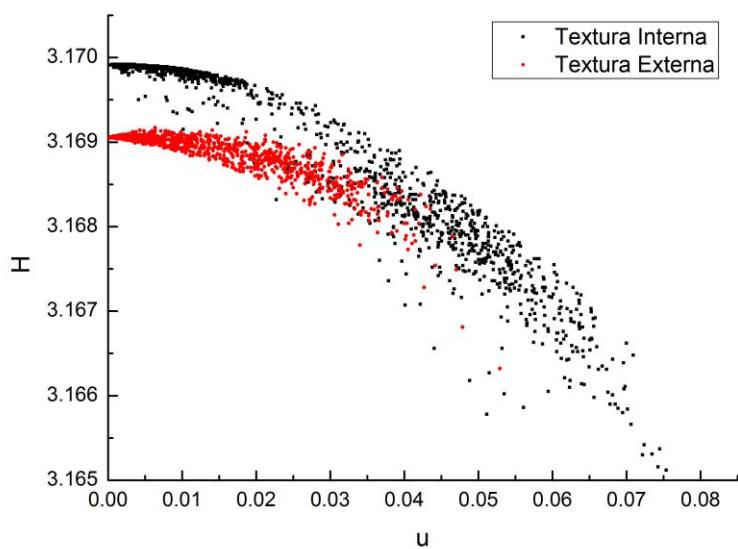


FIGURA 89 ENTROPÍA VS VELOCIDAD PARA LA IMAGEN DE LA FIGURA 88, PARA VECINDARIOS DE 9x9 PIXELES. LOS PUNTOS SE ENCUENTRAN COLOREADOS SEGÚN LA SEGMENTACIÓN DE REFERENCIA.

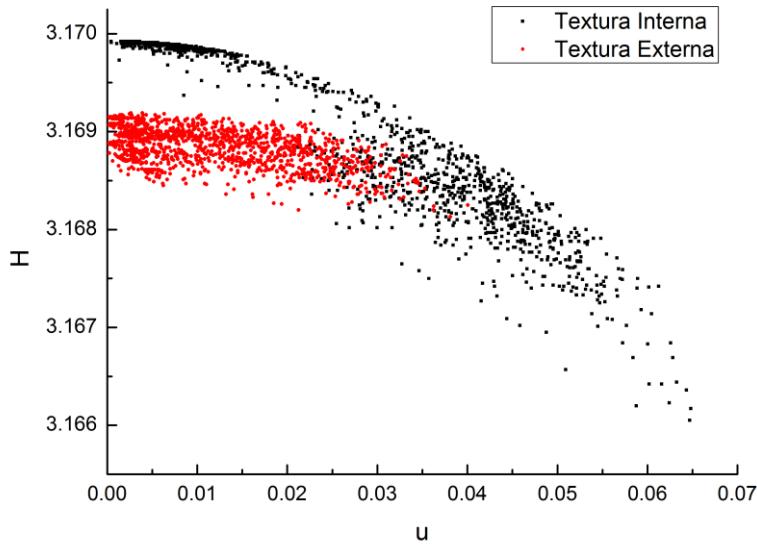


FIGURA 90 ENTROPÍA VS VELOCIDAD PARA LA IMAGEN DE LA FIGURA 88, PARA VECINARIOS DE 11x11 PIXELES. LOS PUNTOS SE ENCUENTRAN COLOREADOS SEGÚN LA SEGMENTACIÓN DE REFERENCIA.

A diferencia de la segmentación angular utilizada en los experimentos anteriores, en este caso hubo que buscar una forma alternativa ya que los parámetros óptimos no incluían al ángulo. A partir de la distribución de puntos correspondiente al par entropía-velocidad, la idea fue introducir una curva que discrimine ambas texturas y que siga la siguiente ecuación:

$$y = (A - ax)^2$$

ECUACIÓN 32 PARÁBOLA DE DISCRIMINACIÓN ENTRE TEXTURAS

Para segmentar adecuadamente la imagen, se realizó un experimento utilizando el índice *AOM* y una segmentación de referencia (construida manualmente). La prueba

consistió en ejecutar el algoritmo y calcular el índice AOM cuando $A=3.1699$ para diferentes valores de a , coeficiente principal de la parábola. Observando el gráfico de la Figura 91, el mejor ajuste de la segmentación se logró para valores de $a=0.94$ estudiando un vecindario de 9×9 pixeles en la imagen. En la Figura 92 se muestra la segmentación final.

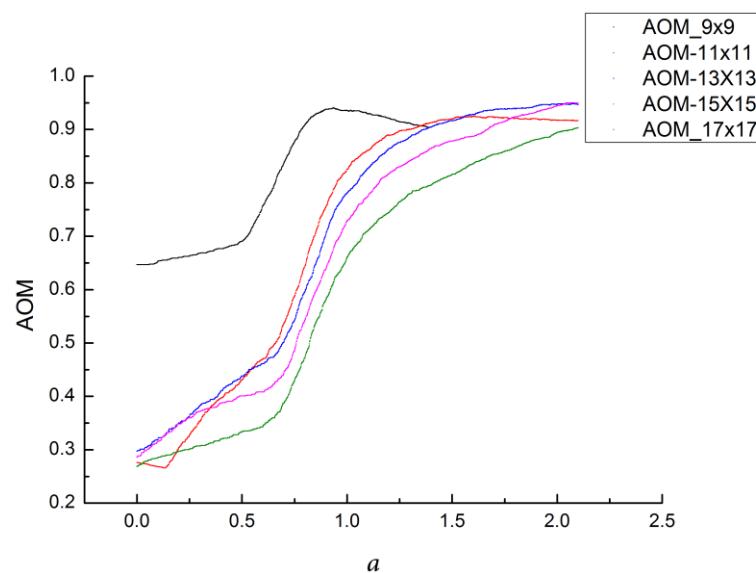


FIGURA 91 SELECCIÓN DEL TAMAÑO DEL VECINDARIO Y DE A, COEFICIENTE PRINCIPAL DE LA PARÁBOLA.



FIGURA 92 SEGMENTACIÓN DE REFERENCIA (IZQUIERDA), SEGMENTACIÓN DE LA FIGURA 88 CORRESPONDIENTE AL $A=0.94$, AOM MÁXIMO PARA VECINDARIOS DE 9×9 PIXELES (DERECHA).

6.7 CASO 7: IMAGEN CON TEXTURA NATURAL

En última instancia se estudió la imagen digital natural mostrada en la Figura 93. Se aplicó la metodología propuesta observando que el tamaño del vecindario guarda relación directa con la discriminación de ambas texturas. En la Figura 94, al graficar el par de variables macroscópicas entropía y velocidad, es posible percibir que la separación entre ambas texturas se hace más notoria al adoptar tamaños de vecindario mayores.



FIGURA 93 - IMAGEN NATURAL COMPUESTA POR TEXTURA DE PINO Y EUCALEPTUS (IZQUIERDA), MASCARA QUE DISTINGUE EL PINO (NEGRO) DEL EUCALEPTUS (BLANCO) (DERECHA).

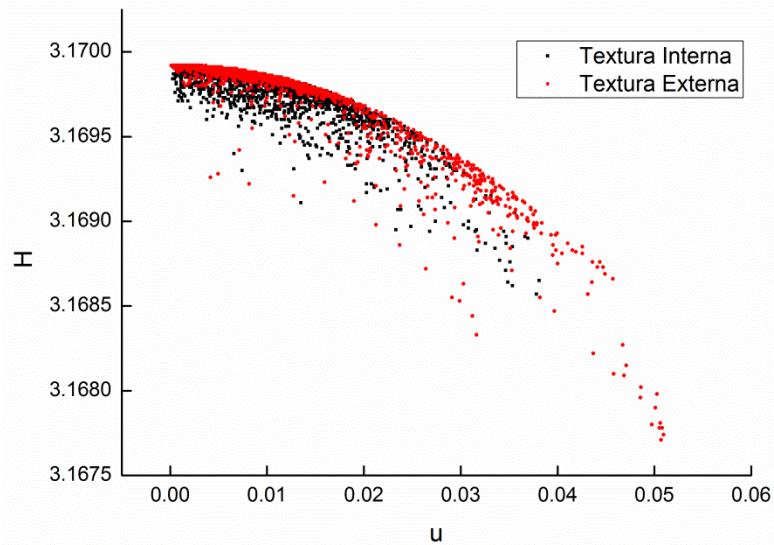


FIGURA 94 - ENTROPÍA VS VELOCIDAD PARA LA IMAGEN DE LA FIGURA 93 USANDO UN VECINDARIO DE 9x9. LOS PUNTOS SE ENCUENTRAN COLOREADOS SEGÚN LA SEGMENTACIÓN DE REFERENCIA.

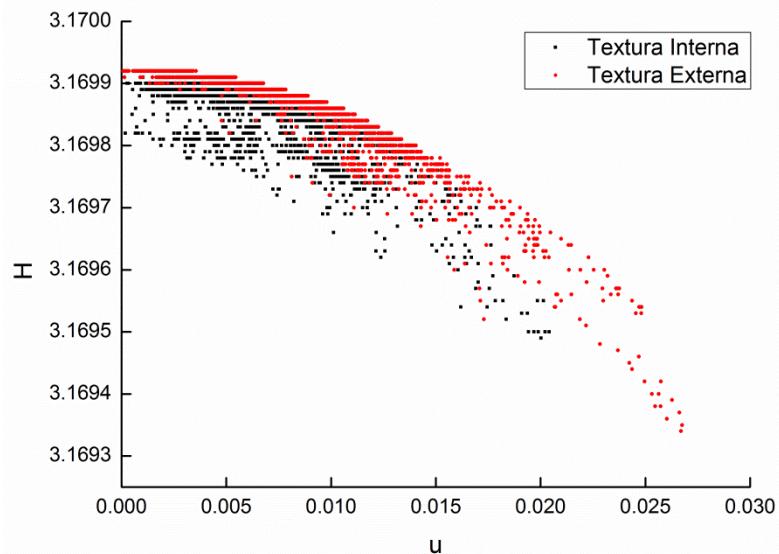


FIGURA 95 ENTROPÍA VS VELOCIDAD PARA LA IMAGEN DE LA FIGURA 93 USANDO UN VECINDARIO DE 17x17. LOS PUNTOS SE ENCUENTRAN COLOREADOS SEGÚN LA SEGMENTACIÓN DE REFERENCIA.

Nuevamente, se segmenta la imagen natural utilizando el índice AOM y una segmentación de referencia (ver Figura 93 y Figura 96). En este experimento, se calculó el índice AOM cuando $A=3.16992$ para diferentes valores de a , coeficiente principal de la parábola. Observando el gráfico de la Figura 96, el mejor ajuste de la segmentación se logró para valores de $A=0.738$ estudiando un vecindario de 17x17 pixeles en la imagen. En la Figura 97 se muestra la segmentación final.

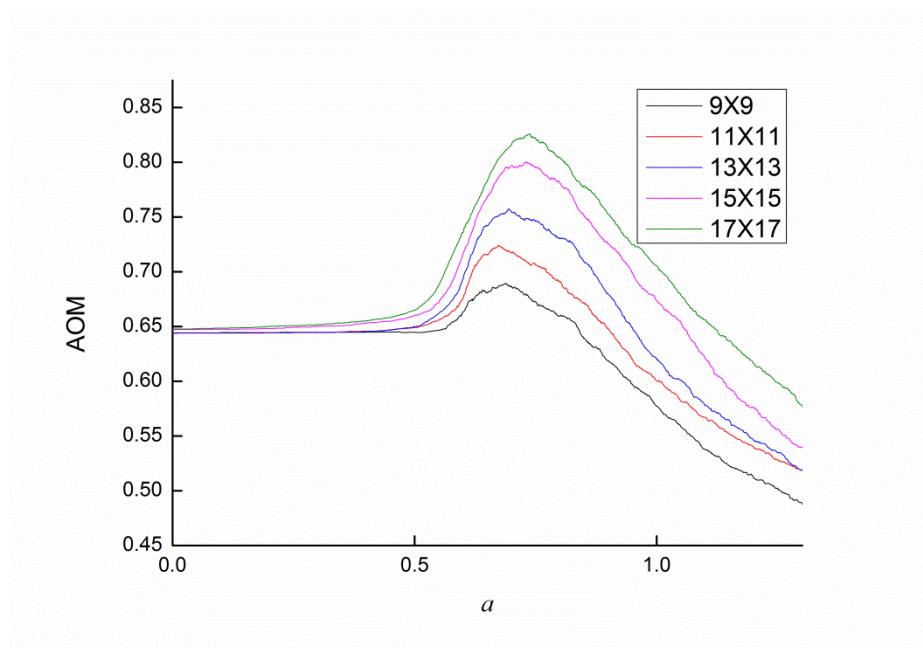


FIGURA 96 AOM VS COEFICIENTE PRINCIPAL DE LA PARÁBOLA ($A=3.16992$) PARA DISTINTOS VECINDARIOS.



FIGURA 97 SEGMENTACIÓN DE LA FIGURA 93 CON TAMAÑO DE VECINDARIO 9, 11, 13, 15 Y 17 PIXELES CADA LADO.

Una nueva imagen estudiada involucra sólo texturas naturales de un bosque de pinos. Puede apreciarse que, en el sector superior derecho de la imagen, se incorporó a la imagen una textura de pinos aún más densificada cuya forma coincide con una herradura

invertida (ver Figura 98 izquierda). Se repitió la metodología planteada para el análisis de esta nueva imagen. En la Figura 99 se graficó el par de variables macroscópicas entropía y velocidad con un vecindario de 17×17 pixeles. Nótese que, las texturas están lo suficientemente dispersas por la parábola $y = 3.16982 - ax^2$ cuya segmentación puede observarse en la Figura 98 (derecha).

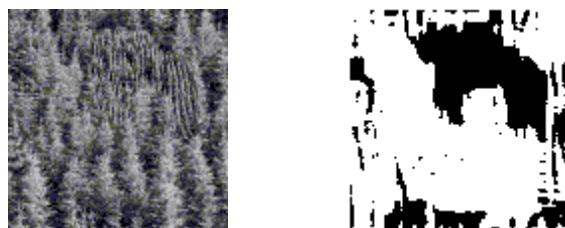


FIGURA 98 IMAGEN NATURAL COMPUESTA POR TEXTURAS A DISTINTA RESOLUCIÓN (IZQUIERDA). SEGMENTACIÓN LOGRADA POR EL ALGORITMO (DERECHA).

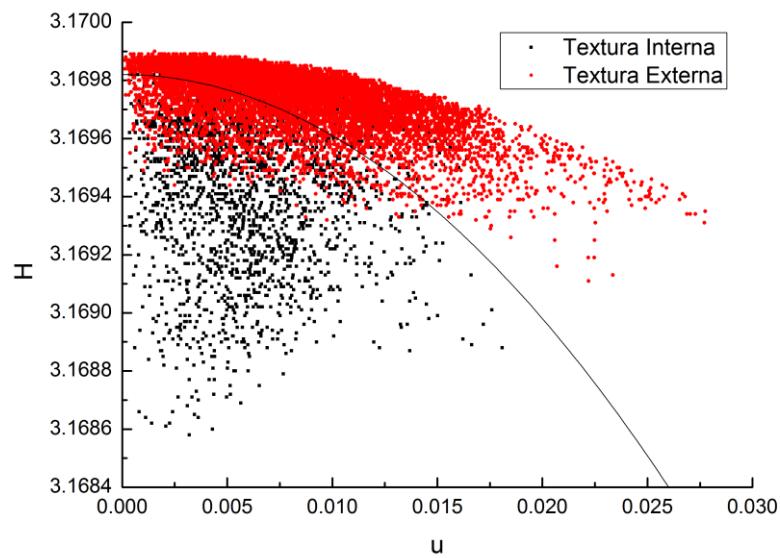


FIGURA 99 ENTROPIA VS VELOCIDAD PARA LA IMAGEN DE LA FIGURA 98. ARRIBA: VECINDARIO DE 17×17 PIXELES. LOS PUNTOS SE ENCUENTRAN COLOREADOS SEGÚN LA SEGMENTACIÓN DE REFERENCIA.

6.8 LIMITACIONES Y ALTERNATIVAS

Si bien el método aquí presentado muestra resultados satisfactorios para los casos estudiados, aún posee limitaciones que deben ser exploradas. Por un lado, el tamaño del patrón que se repite en la textura es de crucial importancia para la correcta detección y posterior segmentación de las figuras que componen una imagen. Si bien el algoritmo ha demostrado robustez frente a la presencia de ruido, es muy sensible a variaciones en el tamaño del vecindario. Es decir, si se estudian texturas cuyo patrón tiene un tamaño de 7×7 , con un vecindario de 3×3 , el promediado presentado al principio de esta sección no resultará efectivo, haciendo que toda la detección fracase.

Esta limitación está relacionada con la aplicación del algoritmo a imágenes reales, especialmente aquellas de carácter natural, como plantas y vegetación en general. En dichas imágenes, el tamaño del patrón de repetición no se conoce de antemano y hasta puede no ser regular (diferentes tamaños y ubicaciones), haciendo que el comportamiento del algoritmo no sea eficiente. La solución que hemos encontrado a este problema se basa en probar con múltiples vecindarios y seleccionar sólo aquellos que maximicen la calidad de la segmentación. Aún así, es posible que exista algún método más efectivo para evitar realizar la búsqueda exhaustiva del vecindario más apropiado.

7. IMPLEMENTACIÓN COMPUTACIONAL

En la siguiente sección se discutirán las principales decisiones arquitectónicas y de diseño adoptadas para la implementación de este trabajo. En primer lugar se hará mención al proceso de desarrollo y la metodología empleada. Seguidamente se explicarán los principales componentes del sistema, las estructuras de datos utilizadas y finalmente algunas consideraciones respecto al algoritmo de simulación.

7.1 PROCESO DE DESARROLLO

Debido a las características cambiantes de los requerimientos el sistema fue desarrollado de forma iterativa, incrementando sus capacidades a medida que surgían los requerimientos (no se conocía de antemano cuáles iban a ser el conjunto de experimentos y las características del sistema que estos iban a necesitar). La elección de este tipo de ciclo de vida del sistema permitió flexibilidad a la hora de las modificaciones, pero también requirió de un *refactoreo* continuo de la estructura general del programa.

En un principio sólo se contó con un pequeño módulo capaz de llevar a cabo la simulación lattice-Boltzmann, y fueron incorporándose al sistema otras características conforme los nuevos experimentos exigían más capacidades. Las siguientes secciones exhiben algunas de las vistas del sistema y ahondan sobre sus particularidades.

7.2 DISEÑO DEL SISTEMA

Dada que la simulación lattice-Boltzmann para el transporte de luz es, básicamente, una multiplicación de matrices, el sistema a construir debía ser capaz de procesar grandes volúmenes de datos y de realizar multiplicaciones en tiempos razonables. Además, fue preciso tener en cuenta que la aplicación no tenía como fin convertirse en un software de uso habitual para la detección y segmentación de texturas, sino que debía servir a una tarea puramente investigativa que implicaría la realización reiterada de experimentos.

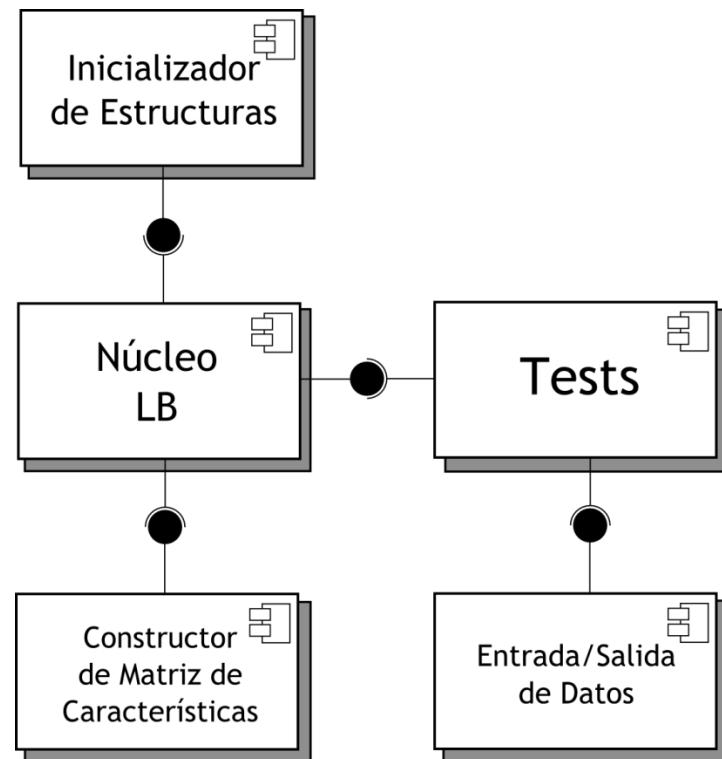


FIGURA 100 DIAGRAMA DE COMPONENTES DEL SISTEMA

Por estas razones, el sistema fue implementado en C++, con la utilización de algunas de las herramientas provistas por el framework Qt, (<http://qt.nokia.com/products/>). Como es sabido, este lenguaje, además de ofrecer una performance sólo comparable a la de lenguajes como Fortran, provee compatibilidad con lenguajes de muy alto nivel, como C# y Java, aspecto interesante si se evalúa la posibilidad de crear aplicaciones de usuario que utilicen el software aquí presentado. Además, C++ es un lenguaje de propósito general, que permite tanto la programación imperativa como la orientada a objetos, y que ha sido extensamente utilizado para el cálculo numérico y la computación científica.



FIGURA 101 FLUJO DEL SISTEMA

La Figura 100 muestra los componentes del sistema y las interacciones entre ellos. El módulo principal (Núcleo LB) contiene aquella funcionalidad relacionada estrechamente con la simulación lattice-Boltzmann y el cálculo de las variables características (dependientes de la matriz de distribución f_i). Entre sus principales responsabilidades se encuentran:

- Llevar a cabo la simulación LB
- Controlar la convergencia del método
- Calcular los valores de las variables macroscópicas una vez alcanzada la convergencia.

El núcleo utiliza los servicios provistos por el módulo Inicializador de estructuras, que las crea en memoria y las carga con los valores indicados por la configuración, y el módulo constructor de la matriz de características o material. Este módulo tiene por objetivos:

- Procesar la imagen sobre la que se realiza la detección
- Crear la matriz de materiales (M_{ij}) a partir de los datos obtenidos de la imagen, y el método indicado en la configuración (Suma, Producto, Resta, etc.).
- Normalizar la matriz (M_{ij}) para que sea estocástica.
- Administrar los valores de difusión aplicados al material.

El módulo de entrada y salida de datos se ocupa de subir a memoria la imagen que es indicada y de guardar los archivos de datos en disco con el formato especificado. Este módulo es capaz de:

- Almacenar cualquiera de los datos devueltos por la simulación, como combinación de varios o en un mismo archivo.
- Almacenar los datos en formato *vtk*.

Ya que para algunos de los test se necesitó que la salida fuera graficada por el kit de herramientas de visualización *VTK* (www.vtk.org/), en este módulo fueron implementados algoritmos para escribir archivos en formatos compatibles con el kit.

Finalmente, en el módulo Tests, se concentraron las implementaciones de todos los experimentos que fueron desarrollados en las Secciones 5 y 6. Este componente realiza la configuración de los parámetros del algoritmo y comunica los resultados al módulo de Entrada/Salida, para que puedan ser almacenados.

7.3 ESTRUCTURAS DE DATOS PRINCIPALES

Las estructuras de datos utilizadas, son las necesarias para implementar computacionalmente las ecuaciones de lattice-Boltzmann desarrolladas en la Sección 3.

7.3.1 LÁTICE

El látice (grilla) es una matriz que almacena el valor lumínico (cantidad de fotones) de cada celda de la grilla (f_i). Se utilizó un arreglo del lenguaje C, de 3 dimensiones $[nx, ny, l]$, donde nx, ny corresponden al tamaño de la imagen o de la región de estudio, y l la cantidad de vecinos de una celda, que en este caso es igual a 9. Para obtener la mayor precisión posible, se trabajó con el tipo *Double*.

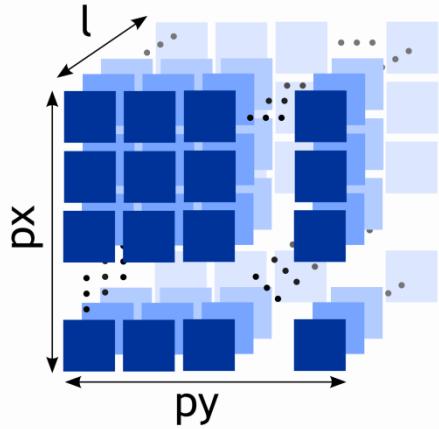


FIGURA 102 REPRESENTACIÓN DE LA MATRIZ DE DENSIDAD f_i

7.3.2 MATRIZ DEL MATERIAL

La matriz del material es la encargada de almacenar la información sobre la imagen. Se utilizó un arreglo de C de 4 dimensiones $[nx, ny, l, l]$, donde nx, ny corresponden al tamaño de la imagen o de la región de estudio, y l la cantidad de vecinos de una celda, que en este caso es igual a 9.

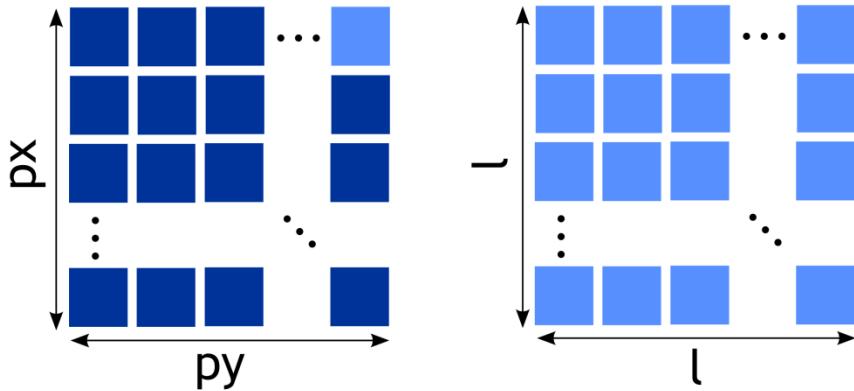


FIGURA 103 REPRESENTACIÓN DE LA MATRIZ DE CARACTERÍSTICAS M_{ij}

Esta matriz puede pensarse como si en cada celda de las primeras dos dimensiones (correspondientes a los píxeles originales de la imagen) se encuentra encontrara otra matriz de dos dimensiones, con las “probabilidades” de que un fotón ingrese por la dirección l_i y salga por la l_j . Para obtener la mayor precisión posible, se trabajó con el tipo *Double*.

7.4 ALGORITMO DE SIMULACIÓN LATTICE-BOLTZMANN

Mediante simulación computacional, se generan los sucesivos pasos de la Ecuación 4 hasta que la variación de la distribución $f_i(\vec{x}, t)$ sea cercana a cero (convergencia). Compuesto únicamente por una multiplicación entre matrices, el modelo LBM para simulación lumínica resulta mucho más sencillo (y menos complejo computacionalmente) que su par de fluidos.

```
simularLB()
{
    While not(converge(F,Fsig))
    {
        Fsig = multiplicarMxF();
        F=Fsig;
    }
}
```

FIGURA 104 PSEUDOCÓDIGO DEL ALGORITMO

La Figura 104 muestra el pseudocódigo del algoritmo. Como es posible observar, al algoritmo está constituido únicamente por la multiplicación de matrices y el método de convergencia. En realidad, esta es una simplificación, que omite cuestiones como la inclusión de los términos de fuente, las restricciones impuestas por las condiciones de entorno, y algunos otros puntos también requeridos por el algoritmo.

8. CONCLUSIONES Y TRABAJOS FUTUROS

En esta tesis de grado se ha logrado principalmente, la aplicación de un algoritmo normalmente utilizado para la simulación de fluidos, al área de detección de texturas en imágenes digitales. Este hecho es completamente novedoso ya que el método resulta, en principio, capaz de detectar texturas sencillas en escala de gris, utilizando parámetros calculados sobre una distribución de densidades obtenidas de la simulación LB.

Por otro lado, se ha encontrado una manera de representar texturas bidimensionales como un medio capaz de interactuar con luz de manera simulada y que puede ofrecer información relevante para la distinción de las texturas que la componen. Los experimentos realizados con las diferentes variables macroscópicas han resultado satisfactorios, aunque es necesario continuar sistemáticamente con esta tarea, para lograr una óptima calibración que consolide al método como una herramienta poderosa para la detección y caracterización de texturas.

Una variedad de líneas de investigación pueden ser comenzadas a partir de los resultados de este trabajo, ya que cada uno de los experimentos tiene aún muchas variables inexploradas. Solamente algunos de ellos fueron aquí abordados en profundidad, como el *Vecindario Promediado*, que resultó ser de extrema utilidad en el proceso de detección. Inclusive el cálculo sencillo de valores como la entropía, es capaz de detectar bordes en imágenes básicas, demostrando que un análisis más profundo de estas operaciones pueden aportar mucho aún a los métodos estadísticos de detección de texturas.

A futuro, se prevé realizar pruebas de mayor tamaño que contengan texturas diferentes para estudiar el comportamiento del método y lograr que el algoritmo sea capaz de

caracterizarlas. Además, deben refinarse los procesos de extracción de características, para que estos se ajusten más a las necesidades de la detección. Además, si bien el método presentado en la Sección 3.4 ofrece buenos resultados, aún pueden encontrarse otras estrategias que solucionen el problema de la normalización por filas de la matriz de características, y que puedan funcionar incluso mejor que lo expuesto en este trabajo.

Por otro lado, aún resta experimentar con la respuesta del algoritmo cuando se introducen fuentes de luz en combinación con diferentes condiciones de contorno. Si bien en las simulaciones efectuadas, las condiciones de contorno empleadas son de tipo periódicas, a futuro podrían realizarse pruebas con nuevas configuraciones que tengan en cuenta condiciones del tipo de fijaciones de variables y/o flujos. La inclusión de este tipo de comportamiento puede también ayudar en el proceso de caracterización.

Finalmente, una vez superada la etapa de análisis del comportamiento del algoritmo, pueden aún ser realizadas una mayor cantidad de pruebas con texturas reales, provenientes de la naturaleza (forestación, cultivos, frutos, tejidos corporales, etc.) o realizadas por el hombre (vehículos, trazado urbano, etc.). Con esos nuevos experimentos será posible determinar el verdadero alcance de las capacidades de los algoritmos de lattice-Boltzmann a la detección de texturas. Del estudio y perfeccionamiento de este método puede surgir, a futuro, la posibilidad de detección de texturas que aún hoy constituyen un problema para los algoritmos disponibles.

REFERENCIAS

Abdel-Dayem A. DETECTION OF ARTERIAL LUMEN IN SONOGRAPHIC IMAGES BASED ON ACTIVE CONTOURS AND DIFFUSION FILTERS, Image Analysis and Recognition (ISBN 978-3-642-13774-7), pp. 120-130, Berlin, 2010.

Castellano G , Bonilha L, Li L.M, Cendes F. TEXTURE ANALYSIS ON MEDICAL IMAGES. Clinical Radiology, Volume 59, Issue 12, (ISSN 59 1061-1069). 2004.

Chopard B, Dupuis A, Masselot A, Luthi P. CELLULAR AUTOMATA AND LATTICE BOLTZMANN TECHNIQUES: AN APPROACH TO MODEL AND SIMULATE COMPLEX SYSTEMS, Advances in complex systems, 5(2):103-246, Ginebra, 2002.

Chen S, Doolen G.D. LATTICE BOLTZMANN METHOD FOR FLUID FLOWS, Annual review of fluid mechanics, 30:329-364, 1998.

Constantin P, Foias C. NAVIER-STOKES EQUATIONS, University of Chicago Press (ISBN 0-226-11548-8), Chicago, 1988.

Currie IG. FUNDAMENTAL MECHANICS OF FLUIDS (ISBN 0-07-014950-X), Albuquerque, 1974.

del Fresno M, Vénere M, and Clausse A. A COMBINED REGION GROWING AND DEFORMABLE MODEL METHOD FOR EXTRACTION OF CLOSED SURFACES IN 3D CT AND MRI SCANS, Computerized Medical Imaging and Graphics, 33(5):369-376, 2009.

Engel K. REAL-TIME VOLUME GRAPHICS, pp. 112-114, 2006.

Epifanio, I. DESCRIPCIÓN DE TEXTURAS. APLICACIONES A SU COMPRENSIÓN Y CLASIFICACIÓN. Tesis Doctoral, Universitat de Valencia, Dpto. de Estadística e Investigación Operativa, Facultad de Matemáticas. Valencia, 2002.

Geist R, Rasche K, Westall J, Schalkoff R. LATICCE-BOLTZMANN LIGHTING, Eurographics Symposium on Rendering, pp. 355-362, Suecia, 2004.

Glassner A. AN INTRODUCTION TO RAY TRACING. Morgan-Kaufmann (ISBN 0-12-286160-4), San Francisco, 1989.

González Rafael C., Richard E. Woods. TRATAMIENTO DIGITAL DE IMÁGENES. Ediciones Díaz de Santos, 1996.

Hájek M, Dezortova M, Materka A, Lerski R. TEXTURE ANALYSIS FOR MAGNETIC RESONANCE IMAGING. Med4Publishing s.r.o., Czech Republic. 2006.

Hernández E. y Herrmann J. CELLULAR AUTOMATA FOR ELEMENTARY IMAGE ENHANCEMENT. Graphical Mode and Image Processing, 58(1):82-89, 1996.

Kassner, A. Thornill, R. E. TEXTURE ANALYSIS: A REVIEW OF NEUROLOGIC MR IMAGING APPLICATIONS. AJNR Am J Neuroradiol 31: 809-16, 2010.

Maso G, Bulant C, D'Amato J, Clausse A, Pomarico J, Iriarte D, Di Rocco H. AUTÓMATA DE LATTICE BOLTZMANN PARA MODELAR LA DIFUSIÓN ÓPTICA EN MATERIALES TRANSLÚCIDOS, 39 Jornadas de informática e investigación operativa (ISSN 1850-2849), pp. 1860-1867, Buenos Aires, 30 de agosto de 2010.

Padma, A. Sukanesh, R. AUTOMATIC CLASSIFICATION AND SEGMENTATION OF BRAIN TUMOR IN CT IMAGES USING OPTIMAL DOMINANT GRAY LEVEL RUN LENGTH TEXTURE FEATURES. International Journal of Advance Computer Science and Applications, 2(10), 2011.

Perona P, J. SCALE-SPACE AND EDGE DETECTION USING ANISOTROPIC DIFFUSION. Proceedings of IEEE Computer Society Workshop on Computer Vision, pp. 16–22, November 1987.

Pichon E, Tannenbaum A, Kikinis R. A STATISTICALLY BASED SURFACE EVOLUTION METHOD FOR MEDICAL IMAGE SEGMENTATION: PRESENTATION AND VALIDATION, Medical Image Computing and Computer -Assisted Intervention - MICCAI 2003 (ISBN 978-3-540-20464-0), pp. 711-720, Berlin, 2003.

Popovici A. y Popovici D. CELLULAR AUTOMATA IN IMAGE PROCESSING. Fifteenth International Symposium on Mathematical Theory of Networks and Systems, 2002.

Preston k., Duff M.J.B, Levialdi S., Norgren P.E. y Toriwaki J. BASIS OF CELULAR LOGIC WITH SOME APPLICATIONS IN MEDICAL IMAGE PROCESSING. Proceedings of the IEEE, 67(5): 629-639, 1979.

Rodríguez J. L. ESTADO ACTUAL DE LA REPRESENTACIÓN Y ANLISIS DE TEXTURA EN IMÁGENES. Centro de Aplicaciones de Tecnologías de Avanzada. 2008.

Rosin P.L. TRAINING CELLULAR AUTOMATA FOR IMAGE PROCESSING. IEEE Transaction on image processing, 15(7): 2076-2087, 2006.

Shapiro G, Stockman G.C. COMPUTER VISION, Upper Saddle River, 2001.

Wongthanavasu S. y Sadananda R. A CA-BASED EDGED OPERATOR AND ITS PERFORMANCE EVALUATION. Journal of Visual Communication and Image Representation, 14(2): 83-96, 2003.