

# CLASIFICACIÓN DE IMÁGENES DE SATÉLITE MEDIANTE AUTÓMATAS CELULARES



GRUPO DE INVESTIGACIÓN DE INFORMÁTICA APLICADA  
DEPARTAMENTO DE LENGUAJES Y COMPUTACIÓN  
UNIVERSIDAD DE ALMERÍA

## Diploma de Estudios Avanzados

Presentado por

A. Moisés Espínola Pérez

Tutelado por

Dr. D. Luis F. Iribarne Martínez  
Profesor Titular de Universidad  
Dpto. Lenguajes y Computación  
Universidad de Almería

Con la colaboración de

Dra. Dña. Rosa M. Ayala Palenzuela  
Profesora Titular de Universidad  
Dpto. Lenguajes y Computación  
Universidad de Almería

Dr. D. Saturnino Leguizamón Avaca  
Profesor Titular de Universidad  
Dpto. Electrónica  
UTN de Mendoza (Argentina)

ALMERÍA, 1 DE MARZO DE 2011



TRABAJO DE INVESTIGACIÓN

CLASIFICACIÓN DE IMÁGENES DE SATÉLITE  
MEDIANTE AUTÓMATAS CELULARES

A. MOISÉS ESPÍNOLA PÉREZ

Este documento ha sido generado  
con la versión LaTeX2e de L<sup>A</sup>T<sub>E</sub>X

Clasificación de imágenes de satélite mediante Autómatas Celulares

A. Moises Espínola Pérez  
Grupo de investigación de Informática Aplicada  
Departamento de Lenguajes y Computación  
Universidad de Almería  
Almería, 1 de Marzo de 2011  
<http://www.ual.es/personal/moises.espinola/>



*A mis abuelos.*



## AGRADECIMIENTOS

Los últimos años de mi vida han sido totalmente fascinantes desde el punto de vista profesional, ya que he podido participar activamente en diversos proyectos de investigación en la Universidad de Almería, y he podido aplicar los conocimientos teóricos y prácticos adquiridos durante mis años de estudios universitarios a la resolución de problemas de distinta índole y complejidad. Desde el punto de vista personal, esta experiencia ha supuesto una intensa y emocionante aventura en la que he conocido a personas cuya profesionalidad y forma de ser dejarán para siempre una profunda huella en mi memoria. A continuación quiero hacer una mención especial de aquellas personas que más han influido en mi labor de investigación.

En primer lugar quiero mencionar a Luis Iribarne, mi verdadero maestro y mentor, y deseo agradecerle por un lado la gran oportunidad que me ha ofrecido al invitarme a participar en esta apasionante aventura, y por otro lado la enorme confianza que ha depositado en mí en numerosas ocasiones. Luis, es un honor tenerte como tutor de este trabajo de investigación.

En segundo lugar deseo mencionar a Rosa Ayala, no sólo por haberme instruido en los aspectos fundamentales de la ciencia de la teledetección, sino también por haberme impregnado positivamente con su personalidad fuerte y optimista. Rosa, gracias de corazón por tus palabras de aliento en los momentos más difíciles.

En tercer lugar quiero mencionar a Saturnino Leguizamón, persona de naturaleza extremadamente sabia e inquieta, por la formación que me ha ofrecido en las técnicas de autómatas celulares. Saturnino, muchas gracias por tus sabios consejos.

También quiero destacar el gran apoyo que siempre he recibido por parte de los Hermanos de las Escuelas Cristianas de La Salle, concretamente de los hermanos directores generales del Colegio La Salle-Virgen del Mar de Almería, Inocencio Lorite Requena y Celerino Sanz Granada, que me han permitido asistir en período docente a los congresos en los que he tenido que impartir alguna conferencia sobre mi tema de investigación. Sin su comprensión, este trabajo no habría salido a la luz.

Además quiero destacar los consejos ofrecidos por Amparo Fúster Sabater, miembro del CSIC de Madrid, así como sus palabras de ánimo en mi futura labor investigadora, que recibí con agrado en una pequeña ciudad de Italia llamada Ascoli Piceno.

También quisiera dar las gracias a todos aquellos compañeros de la Universidad de Almería que me han apoyado y aconsejado, como por ejemplo Julio Gómez, Raúl Baños, Antonio López Márquez y José Antonio Álvarez Bermejo.

Y por último, y no menos importante, quiero dar las gracias a mis padres por haberme dado la educación y formación necesaria para ser lo que soy en la actualidad. Todo lo que soy, se lo debo a ellos.

Este trabajo ha sido parcialmente financiado por la UE (FEDER) y el Ministerio de Ciencia e Innovación de España con la subvención del proyecto I+D TIN2007-61497 “SOL-ERES: un sistema de información espacio-temporal para la gestión medioambiental basado en redes neuronales, agentes y componentes software” del grupo de investigación de Informática Aplicada (TIC-211) del departamento de Lenguajes y Computación de la Universidad de Almería.

Moisés Espínola Pérez  
Grupo de investigación de Informática Aplicada  
Departamento de Lenguajes y Computación  
Universidad de Almería  
ALMERÍA, 1 DE MARZO DE 2011



# Contenido

PRÓLOGO . . . . .	xv
<b>1. PERÍODO DOCENTE</b>	<b>3</b>
1.1. PROGRAMA DE DOCTORADO . . . . .	5
1.2. CURSOS DE DOCTORADO . . . . .	6
1.2.1. Modelado de sistemas de información . . . . .	6
1.2.2. Metodologías de Compresión, Restauración y Reconstrucción de im- agen. . . . .	7
1.2.3. Optimización, heurística y multiobjetivo. Estrategias de paralelización	9
1.2.4. Algoritmos de optimización global. Estrategias paralelas. . . . .	10
1.2.5. Problemas matriciales en sistemas multiprocesador . . . . .	11
1.2.6. Balanceo dinámico de carga en cluster computing. Aplicación en To- mografía Computerizada . . . . .	12
1.3. RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .	13
<b>2. TELEDETECCIÓN</b>	<b>17</b>
2.1. INTRODUCCIÓN A LA TELEDETECCIÓN . . . . .	19
2.1.1. Definición de teledetección . . . . .	19
2.1.2. Historia de la teledetección . . . . .	20
2.1.3. Elementos básicos de la teledetección . . . . .	22
2.1.4. Aspectos físicos de la teledetección . . . . .	23
2.2. CARACTERÍSTICAS DE LAS IMÁGENES DE SATÉLITE . . . . .	24
2.2.1. Formato de las imágenes de satélite . . . . .	24
2.2.2. Resolución de las imágenes de satélite . . . . .	28
2.2.3. Errores en la captación de imágenes . . . . .	29
2.2.4. Clasificación de imágenes de satélite . . . . .	29
2.3. ALGORITMOS DE CLASIFICACIÓN NO SUPERVISADOS . . . . .	32
2.3.1. Clasificador k-medias . . . . .	32
2.3.2. Clasificador isodata . . . . .	33
2.4. ALGORITMOS DE CLASIFICACIÓN SUPERVISADOS . . . . .	35
2.4.1. Clasificador mínima distancia . . . . .	35
2.4.2. Clasificador paralelepípedos . . . . .	36
2.4.3. Clasificador máxima verosimilitud . . . . .	37
2.5. RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .	38
<b>3. AUTÓMATAS CELULARES</b>	<b>41</b>
3.1. INTRODUCCIÓN A LOS AUTÓMATAS CELULARES . . . . .	43
3.1.1. Definición conceptual de autómatas celular . . . . .	43
3.1.2. Definición formal de autómatas celular . . . . .	44
3.1.3. Clasificación de los autómatas celulares . . . . .	45
3.2. APLICACIONES DE LOS AUTÓMATAS CELULARES . . . . .	46

3.2.1.	Autómatas celulares unidimensionales. Notación de Stephen Wolfram	46
3.2.2.	Autómatas celulares bidimensionales. Juego de la vida de Conway	50
3.2.3.	Autómatas celulares aplicados a la teledetección	51
3.3.	<b>CARACTERIZACIÓN DE TEXTURAS MEDIANTE AUTÓMATAS CELULARES</b>	<b>52</b>
3.3.1.	Números decimales de las reglas de Stephen Wolfram	52
3.3.2.	Caracterización de texturas de imágenes-AC	53
3.3.3.	Caracterización de texturas naturales	55
3.4.	<b>RESUMEN Y CONCLUSIONES DEL CAPÍTULO</b>	<b>59</b>
<b>4.</b>	<b>CLASIFICACIÓN CON AUTÓMATAS CELULARES</b>	<b>63</b>
4.1.	PROBLEMAS DE LOS ALGORITMOS DE CLASIFICACIÓN SUPERVISADOS	65
4.1.1.	Inconvenientes de los algoritmos de clasificación supervisados	65
4.1.2.	Cálculo de la calidad del proceso de clasificación	66
4.1.3.	Análisis de los problemas en la clasificación supervisada clásica	68
4.2.	CLASIFICACIÓN DE IMÁGENES DE SATÉLITE CON AUTÓMATAS CELULARES	74
4.2.1.	Trabajo previo existente	74
4.2.2.	Objetivos principales del algoritmo ACA	74
4.2.3.	Algoritmo ACA v1.0: estados y reglas	76
4.3.	DESCRIPCIÓN DEL ALGORITMO ACA	78
4.3.1.	Arquitectura del algoritmo ACA	78
4.3.2.	Pseudocódigo del algoritmo ACA	79
4.3.3.	Simulación del algoritmo ACA	82
4.4.	<b>RESUMEN Y CONCLUSIONES DEL CAPÍTULO</b>	<b>83</b>
<b>5.</b>	<b>RESULTADOS OBTENIDOS</b>	<b>87</b>
5.1.	MEJORA DE LA CALIDAD DE LA CLASIFICACIÓN	90
5.2.	OBTENCIÓN DE UNA CLASIFICACIÓN JERÁRQUICA	91
5.3.	DETECCIÓN DE BORDES, PÍXELES INCIERTOS Y RUIDOSOS	95
5.4.	<b>RESUMEN Y CONCLUSIONES DEL CAPÍTULO</b>	<b>96</b>
<b>6.</b>	<b>APORTACIONES Y TRABAJO FUTURO</b>	<b>99</b>
6.1.	PROYECTOS I+D	101
6.2.	PUBLICACIONES	102
6.3.	TRABAJO FUTURO	104
6.4.	<b>RESUMEN Y CONCLUSIONES DEL CAPÍTULO</b>	<b>105</b>
	<b>PROPUESTA DE ADSCRIPCIÓN A UN ÁREA DE CONOCIMIENTO</b>	<b>III-1</b>
	<b>BIBLIOGRAFÍA</b>	<b>III-1</b>

# Índice de Figuras

2.1.	Elementos básicos de un sistema genérico de teledetección. . . . .	22
2.2.	Espectro electromagnético y técnicas de teledetección. . . . .	23
2.3.	(a) Imagen multiespectral. (b) Imagen hiperespectral. . . . .	24
2.4.	Imagen DEM (Digital Elevation Model). . . . .	25
2.5.	Imagen de satélite en el espacio de características (3 clases distintas). . . . .	25
2.6.	(a) Imagen Landsat-7 de las provincias Almería y Granada (bandas 2, 3 y 4). (b) Imagen Landsat-7 de las provincias Almería y Granada (bandas 1, 2 y 3: color verdadero). . . . .	26
2.7.	(a) Olas de gravedad atmosférica en Nueva Zelanda. (b) Nieve en Gran Bretaña e Irlanda. (c) Tormentas de polvo en el suroeste de Asia. (d) Floración de fitoplancton en Argentina. (Fuente: NASA). . . . .	27
2.8.	Proceso general de clasificación de imágenes de satélite. . . . .	30
2.9.	(Izquierda) Clasificación no supervisada. (Derecha) Clasificación supervisada. . . . .	31
2.10.	(a) Descripción del proceso general de clasificación. (b) Resultados de una clasificación supervisada. (c) Resultados de una clasificación no supervisada. . . . .	32
2.11.	Clasificación mediante el algoritmo mínima distancia. . . . .	35
2.12.	Clasificación mediante el algoritmo paralelepípedos. . . . .	36
2.13.	Clasificación mediante el algoritmo máxima verosimilitud. . . . .	37
3.1.	(a) Vecindad de von Neumann. (b) Vecindad de Moore. (c) Vecindad de Moore extendida. . . . .	44
3.2.	Imagen-AC generada por la regla R94. . . . .	47
3.3.	Imagen-AC generada por la regla R54. . . . .	48
3.4.	Reglas que generan las imágenes-AC más conocidas. . . . .	49
3.5.	Imágenes-AC más conocidas de Stephen Wolfram. . . . .	49
3.6.	(a) Soledad: una celda viva con menos de 2 celdas vivas adyacentes muere. (b) Sobrepoblación: una celda viva con más de 3 celdas vivas adyacentes muere. (c) Reproducción: una celda muerta con 3 celdas vivas adyacentes nace. (d) Estabilidad: una celda viva con dos celdas vivas adyacentes permanece viva. . . . .	51
3.7.	(a) Condiciones iniciales. (b) Iteración 10. (c) Iteración 25. (d) Iteración 50. (e) Iteración 75. . . . .	51
3.8.	(a) Imagen-AC obtenida con la regla R225. (b) Histograma de números decimales obtenidos aplicando el método propuesto. . . . .	54
3.9.	Primitivas de textura correspondientes a la imagen-AC generada por la regla R102. . . . .	54

3.10. (Izquierda) Imagen (256x256) compuesta por 4 sectores de distintas texturas: sectores superiores (R121 y R153), sectores inferiores (R60 y R90). (Derecha) Imagen en escala de grises resultante de la aplicación del método propuesto. . . . .	55
3.11. Arriba: 3 imágenes del álbum Brodatz: D3, D49 y D85. Abajo: imágenes binarias correspondientes. . . . .	56
3.12. Histogramas de primitivas normalizados para la imagen Brodatz binaria D85. (a) HPH (dirección horizontal). (b) VHP (dirección vertical). . . . .	57
3.13. (a) Imágenes-AC obtenidas como resultado de elegir las primitivas más importantes de los histogramas anteriores. Del histograma vertical se han tomado 6 primitivas y del horizontal 3. Aplicando el método propuesto se han obtenido las reglas 207 (horizontal) y 193 (vertical). . . . .	57
3.14. Izquierda: imagen bin3 (256x256). Derecha: aproximación de la imagen bin3 obtenida por la composición de las imágenes-AC R95 (H) y R95 (V) obtenidas usando 6 componentes de cada histograma primitivo (H y V). . . . .	58
3.15. Izquierda: imagen bin49 (256x256). Derecha: aproximación de la imagen bin49 obtenida por la composición de las imágenes-AC R217 (H) y R76 (V) obtenidas usando 5 componentes del HPH y 3 componentes del VPH. . . . .	58
3.16. Izquierda: imagen bin85 (256x256). Derecha: aproximación de la imagen bin85 obtenida por la composición de las imágenes-AC R207 (H) y R193 (V) obtenidas usando 6 componentes del HPH y 3 componentes del VPH. . . . .	58
4.1. Clasificación obtenida mediante trabajo de campo. . . . .	67
4.2. Clasificación obtenida mediante el algoritmo mínima distancia. . . . .	67
4.3. Clasificación obtenida mediante trabajo de campo centrada en los 2 píxeles de estudio. . . . .	70
4.4. Clasificación obtenida mediante el algoritmo mínima distancia centrada en los 2 píxeles de estudio. . . . .	70
4.5. Valores espectrales de 2 píxeles mal clasificados en las 7 capas. . . . .	71
4.6. Imagen de satélite con 1% de ruido. . . . .	73
4.7. Valor espectral de un píxel ruidoso. . . . .	73
4.8. Arquitectura completa del clasificador ACA. . . . .	78
4.9. Evolución del algoritmo ACA con 3 iteraciones: (primera línea) mínima distancia, (segunda línea) paralelepípedos, (tercera línea) máxima verosimilitud. . . . .	82
5.1. Región tomada de la imagen de satélite completa de las provincias de Almería y Granada (España). . . . .	89
5.2. Clasificación jerárquica en 6 capas de fiabilidad. . . . .	91
5.3. Clasificación obtenida en las iteraciones 1-12 del algoritmo ACA. . . . .	92
5.4. Clasificación obtenida en las iteraciones 13-24 del algoritmo ACA. . . . .	93
5.5. Clasificación obtenida en las iteraciones 26-30, 40, 50, 60, 70, 80, 90 y 100. . . . .	94
5.6. Gráfica: distribución del tiempo necesario para clasificar la imagen de estudio mediante 100 iteraciones del algoritmo ACA. . . . .	95



# Índice de Tablas

1.1. Calificación obtenida en los cursos de doctorado. . . . .	6
2.1. Algoritmo de clasificación k-medias. . . . .	33
2.2. Algoritmo de clasificación isodata. . . . .	34
3.1. Evolución del autómata celular unidimensional del ejemplo. . . . .	46
3.2. Adquisición de los números decimales de la regla de 3 celdas. . . . .	53
3.3. Algoritmo de caracterización de texturas naturales mediante autómatas celulares. . . . .	56
4.1. Ventajas e inconvenientes de los algoritmos de clasificación supervisados más usados: mínima distancia, paralelepípedos y máxima verosimilitud. . . . .	66
4.2. Matriz de confusión del algoritmo mínima distancia. . . . .	68
4.3. Lista de 200 píxeles mal clasificados tomados al azar. En negrita se destacan los píxeles que poseen la relación de clases “5, 7”. . . . .	69
4.4. Valores espectrales de 2 píxeles mal clasificados. . . . .	72
4.5. Valores espectrales de las 8 clases seleccionadas. . . . .	72
4.6. Distancia del píxel erróneo a la clase 5. . . . .	72
4.7. Distancia del píxel erróneo a la clase 7. . . . .	72
4.8. Algoritmo ACA principal. . . . .	80
4.9. Algoritmo ACA espectral. . . . .	81
4.10. Algoritmo ACA contextual. . . . .	81
5.1. Matriz de confusión del algoritmo mínima distancia. . . . .	90
5.2. Matriz de confusión del algoritmo ACA. . . . .	90



# PRÓLOGO

La teledetección es una ciencia aplicada que permite la adquisición de información mediante técnicas que no requieren contacto físico con el objeto o área que está siendo observado. Las técnicas de teledetección se pueden utilizar en muchas aplicaciones medioambientales, ayudando a resolver y mejorar los problemas sociales derivados de ellos. Entre el amplio repertorio de aplicaciones de la teledetección podemos encontrar ejemplos que incluyen estudios de calidad del suelo, búsqueda de recursos hídricos, protección del medio ambiente o simulaciones meteorológicas, entre otros.

Los algoritmos de clasificación de imágenes de satélite son una de las técnicas más importantes utilizadas en el ámbito de la teledetección, ya que facilita a los investigadores la interpretación de la gran cantidad de información contenida en sus bandas. En la actualidad existen muchos tipos distintos de algoritmos de clasificación, como por ejemplo el clasificador de mínima distancia, paralelepípedos y máxima verosimilitud. En este trabajo se investiga el desarrollo de un nuevo algoritmo de clasificación de imágenes de satélite basado en autómatas celulares llamado ACA (Algorithm based on Cellular Automata), una técnica que normalmente se ha utilizado para la investigación de sistemas complejos. No hay trabajos previos relacionados con la clasificación de imágenes de satélite mediante autómatas celulares. Este nuevo tipo de clasificador de imágenes de satélite, que mejora en varios aspectos los resultados obtenidos por otros algoritmos clásicos, ha sido validado y experimentado en el marco de los proyectos I+D *SOLERES: un sistema de información espacio-temporal para la gestión medioambiental basado en redes neuronales, agentes y componentes software* (TIN2007-61497) e *iSOLERES: una metodología para la recuperación y explotación de información medioambiental mediante interfaces de usuario evolutivas y cooperativas* (TIN2010-15588), ambos subvencionados por el Ministerio de Ciencia e Innovación de España.

La finalidad principal del proyecto SOLERES consiste en desarrollar investigación aplicada para el diseño e implementación básica de un sistema de información espacio-temporal para la gestión medioambiental, soportado por la aplicación, integración y desarrollo (extensión) de trabajos previos multidisciplinares en I+D del equipo de investigación y de innovación tecnológica en imágenes de satélite, redes neuronales, sistemas cooperativos basados en arquitecturas multiagente y agentes inteligentes y sistemas software basados en componentes comerciales. Este proyecto integra y desarrolla investigación aplicada y multidisciplinar (como en ingeniería del software, ingeniería del conocimiento e inteligencia artificial, y ecología) para el modelado de un sistema de información inteligente en gestión medioambiental. En este proyecto se pretende, por un lado, estudiar la generación automática de mapas ecológicos a partir de información de satélite. La obtención automática de estos mapas puede ser de gran utilidad en actuaciones críticas de gestión medioambiental, como en tareas de predicción/prevenición y de toma de decisiones, en fenómenos naturales de riesgo (p.e., inundaciones, incendios forestales, terremotos). Debido a la diversidad de usuarios finales (p.e., políticos, técnicos, administradores) y al gran volumen

y variado tipo de información (alguna crítica y confidencial), es importante contar con propuestas científico-técnicas reales y prácticas para la construcción de rápidos y eficientes sistemas de explotación de información (en este caso medioambiental), facilitando la interacción hombre-máquina con interfaces de usuario dinámicas que se adapten a los hábitos de los perfiles de los usuarios, y con asesores (agentes) de software inteligentes que medien por los usuarios en los procesos de búsqueda y de explotación de la información y que faciliten las tareas de toma de decisiones (medioambiental), y de predicción/prevenición (que son las más importantes). El sistema de información está siendo definido mediante el paradigma del Trabajo Cooperativo Asistido por Ordenador (CSCW, Computer Supported Cooperative Work) e implementado con tecnología de agentes inteligentes y arquitecturas multiagente. Para el proceso de explotación de datos se está identificando y estructurando el tipo de información susceptible de ser consultada y el tipo de consultas, usando técnicas de descomposición jerárquica (p.e., árboles, recorte y poda) y redes neuronales. Para la futura integración de todas las técnicas adoptadas y desarrolladas durante el proyecto, se realizará un estudio completo de evaluación, interoperabilidad, adaptación, pruebas (individuales y colectivas) y ensamblaje usando técnicas de desarrollo de software basado en componentes COTS para, finalmente, implementar un prototipo del sistema.

El presente documento consta de tres secciones principales: período docente, período investigador y anexos. A continuación se detalla el contenido de cada una de dichas secciones:

- Período docente: corresponde al capítulo 1, donde se ofrece una descripción tanto del programa de doctorado como de cada uno de los cursos realizados durante el período docente.
- Período investigador: abarca desde el capítulo 2 hasta el capítulo 6, ambos incluidos. En estos capítulos se ofrece una introducción a la ciencia aplicada de teledetección (capítulo 2), una descripción de los autómatas celulares (capítulo 3), una descripción del trabajo desarrollado a lo largo del período de investigación tutelado (capítulo 4), una exposición de los resultados obtenidos (capítulo 5) y un resumen de las aportaciones realizadas durante la fase de investigación así como del trabajo futuro que se pretende desarrollar para obtener la tesis doctoral (capítulo 6).
- Anexos: formado por la propuesta de adscripción del trabajo de investigación presentado en este documento a un área de conocimiento y la bibliografía consultada para la elaboración de esta memoria.

---

# CAPÍTULO 1

## PERÍODO DOCENTE

---



# Capítulo 1

## PERÍODO DOCENTE

### Contenidos

---

<b>1.1.</b>	PROGRAMA DE DOCTORADO . . . . .	<b>5</b>
<b>1.2.</b>	CURSOS DE DOCTORADO . . . . .	<b>6</b>
1.2.1.	Modelado de sistemas de información . . . . .	6
1.2.2.	Metodologías de Compresión, Restauración y Reconstrucción de imagen. . . . .	7
1.2.3.	Optimización, heurística y multiobjetivo. Estrategias de paralelización . . . . .	9
1.2.4.	Algoritmos de optimización global. Estrategias paralelas. . . . .	10
1.2.5.	Problemas matriciales en sistemas multiprocesador . . . . .	11
1.2.6.	Balanceo dinámico de carga en cluster computing. Aplicación en Tomografía Computerizada . . . . .	12
<b>1.3.</b>	RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .	<b>13</b>

---





**S**egún las normas reguladoras de los estudios de Tercer Ciclo de la Universidad de Almería para la obtención del Diploma de Estudios Avanzados y la suficiencia investigadora, el doctorando debe realizar un mínimo de 32 créditos en el programa de doctorado al que esté adscrito, distribuidos en dos períodos de la forma siguiente:

- En el período de docencia deberá completar un mínimo de 20 créditos.
- En el período de investigación deberá completar un mínimo de 12 créditos, que se invertirán necesariamente en el desarrollo de uno o varios trabajos de investigación tutelados a realizar dentro del Departamento donde se desarrolle el programa de doctorado.

Este capítulo describe mi período de docencia, y el resto de capítulos del libro expone mi período de investigación. Para describir mi período de docencia, los siguientes apartados ofrecen información detallada tanto del programa de doctorado al que me adscribí, como de los cursos de doctorado que superé.

## 1.1. PROGRAMA DE DOCTORADO

Los estudios realizados pertenecen al programa de doctorado “*Técnicas Informáticas Avanzadas (8021)*” durante el curso académico 2005-06, cuya responsable fue Gracia Ester Martín Garzón (Dpto. de Arquitectura de Computadores y Electrónica). Los departamentos implicados en la elaboración del doctorado son los siguientes:

- Lenguajes y Computación (UAL) (Dpto. responsable).
- Arquitectura de Computadores y Electrónica (UAL).
- Estadística y Matemática Aplicada (UAL).
- Álgebra y Análisis Matemático (UAL).

Durante el programa de doctorado *Técnicas Informáticas Avanzadas*, curso académico 2005-06, se impartieron las siguientes asignaturas:

- Códigos y criptografía.
- Sistemas expertos probabilísticos y razonamiento temporal aproximado.
- Metodologías de compresión, restauración y reconstrucción de imagen.
- Algoritmos numéricos y lenguajes formales.
- Algoritmos de optimización global. Estrategias paralelas.
- Optimización heurística y multiobjetivo. Estrategias de paralelización.
- Introducción a los fractales y al caos.
- Técnicas avanzadas en automática, comunicaciones para control y robótica.
- Métodos de integración de información. Aplicación a datos geográficos.

- Balanceo dinámico de carga en cluster computing. Aplicación en Tomografía Computarizada.
- Bases de datos deductivas y aplicaciones.
- Modelado conceptual del conocimiento en sistemas informáticos.
- Sistemas cooperativos basados en servicios web.
- Problemas matriciales en sistemas multiprocesador.
- Modelado de sistemas de información.

Según la normativa de los estudios de tercer ciclo hay que cursar un mínimo de 20 créditos, y en mi caso cursé un total de 28 créditos (6 cursos de doctorado). La elección de los cursos de doctorado fue motivada por la aplicación que podrían tener sus contenidos para mis futuras investigaciones. Durante el período docente correspondiente al programa de doctorado “*Técnicas Informáticas Avanzadas (8021)*” superé los cursos de doctorado que se muestran en la Tabla 1.1.

Asignatura	Créditos	Calificación
Modelado de sistemas de información	6	Sobresaliente
Metodologías de Compresión, Restauración y Reconstrucción de Imagen	5	Sobresaliente
Optimización heurística y multiobjetivo. Estrategias de paralelización	4	Sobresaliente
Algoritmos de optimización global. Estrategias paralelas	5	Sobresaliente
Problemas matriciales en sistemas multiprocesador	5	Sobresaliente
Balanceo dinámico de carga en cluster computing. Aplicación en Tomografía	3	Sobresaliente

Tabla 1.1: Calificación obtenida en los cursos de doctorado.

Durante el período de investigación he realizado la asignatura “Modelado de sistemas de información” de 12 créditos tutelada por los profesores Luis F. Iribarne Martínez y Jesús Manuel Almendros Jiménez. El resultado de la investigación ha tenido buena acogida por el comité científico internacional ya que ha sido publicado en varios congresos internacionales.

## 1.2. CURSOS DE DOCTORADO

En este apartado se muestra un resumen de cada uno de los 6 cursos de doctorado que he superado durante el período de docencia, ordenados por el grado de relevancia que han tenido en mi trabajo de investigación.

### 1.2.1. Modelado de sistemas de información

**Número de créditos:** 6.

**Profesorado:**

- Antonio Becerra-Terón.
- Jesús Manuel Almendros Jiménez.
- Luis Fernando Iribarne Martínez.

- Nicolás Padilla Soriano.

**Programa del curso:**

Bloque I: Modelado de Sistemas Cooperativos.

- Visión general de los Sistemas Cooperativos.
- La interacción en los grupos a través del ordenador.

Bloque II: Modelado de Servicios Web.

- Agentes software.
- Primera Generación.
- Los Servicios Web. Segunda Generación.

Bloque III: Modelado de Datos Web.

- XML.
- XPath/XQuery.
- RDF/OWL.

**Motivación de la elección del curso:**

La modelización de sistemas de información en la actualidad está orientada a las llamadas nuevas tecnologías. Las nuevas tecnologías surgidas como consecuencia del nacimiento y crecimiento de la Web necesitan de nuevos modelos de diseño. Estos nuevos modelos incluyen entre otros las bases de datos web, los servicios web y los sistemas cooperativos. Además nuevos lenguajes de modelado como UML han sido introducidos con el fin de modelar estos sistemas.

Tradicionalmente la mayoría de las aplicaciones informáticas se han basado en el paradigma en el cual una persona trabaja individualmente para realizar una determinada tarea. Sin embargo, no es habitual esta forma de organización del trabajo. De hecho, el trabajo se organiza generalmente en equipo y cada persona interactúa con el resto para obtener una mayor productividad. Las investigaciones realizadas hasta ahora sobre grupos mediados por ordenador se han centrado en lo que se conoce como sistemas cooperativos (CSCW).

Por otro lado, últimamente podemos comprobar una paulatina consolidación de las técnicas empleadas en la programación web, como WSDL, SOAP y UDDI, y el aumento en el uso de los servicios web para construir sistemas de información basados (parcial o totalmente) en web. Las técnicas basadas en servicios web permiten la descripción (WSDL) y la publicación y localización (UDDI) de servicios específicos que funcionan bajo recursos Internet, facilitando la programación de complejos sistemas cooperativos distribuidos basados en servicios web donde uno o más servicios pueden requerir la presencia de otros para funcionar, tal y como ocurre en DSBC con el ensamblaje de componentes software.

**1.2.2. Metodologías de Compresión, Restauración y Reconstrucción de imagen.**

**Número de créditos:** 5.

**Profesorado:**

- José Jesús Fernández Rodríguez.
- Vicente González Ruiz.

**Programa del curso:**

Bloque I: Compresión de imágenes.

- Introducción.
  - ¿Qué es una imagen?
  - ¿Qué es un vídeo?
  - ¿Por qué comprimir?
  - ¿Qué tipo de compresores existen?
  - ¿Cuáles son sus principales posibilidades?
  - Escalabilidad.
- Compresión de texto.
  - Fundamentos de los compresores de texto.
  - Un ejemplo real: gzip.
  - Compresión de “akiyo” usando gzip.
- Compresión de imagen.
  - Fundamentos de los compresores de imágenes.
  - El formato de imagen PNG (Portable Network Graphics).
  - Compresión de “akiyo” usando PNG.
  - El estándar JPEG (ISO/IEC 10918-1).
  - Calidad de JPEG vs factor de compresión.
  - Compresión de “akiyo” usando JPG.
  - El estándar JPEG 2000 (ISO/IEC 15444-1).
  - JPEG vs JPEG 2000.
  - Compresión de “akiyo” usando JPG 2000.
- Compresión de secuencias de imagen (vídeo).
  - Fundamentos de los compresores de secuencias de imágenes.
  - El estándar MPEG-1 (ISO/IEC 11172).
  - Compresión de “akiyo” usando MPEG-1.
  - El estándar MPEG-2 (ISO/IEC 13818).
  - Compresión de “akiyo” usando MPEG-2.
  - El estándar MPEG-4 (ISO/IEC 14496).
  - Compresión de “akiyo” usando MPEG-4.
- El futuro.
  - En compresión de texto.
  - En compresión de imágenes.

- En compresión de secuencias de imágenes.

Bloque II: Restauración y reconstrucción de imagen.

- Introducción a la transformada de Fourier.
- Filtrado, mejora y restauración de imagen.
- Reconstrucción tomográfica.
- Una aplicación práctica de Reconstrucción 3D.
- Filtrado de Ruido con Anisotropic Nonlinear Diffusion.

**Motivación de la elección del curso:**

Este curso prepara al alumno para conocer y comprender los diferentes protocolos que se utilizan para la compresión y restauración tanto de imágenes como de vídeo. Al tratarse de una asignatura que profundiza en los procesos de codificación y procesamiento de imágenes, se trata de una materia importante para aquellos alumnos que deseen especializarse en éstas líneas, como es mi caso.

El curso se dividió en dos grandes bloques: comprensión de imágenes y, restauración y reconstrucción de imágenes.

En el primer bloque se analizaron los diferentes protocolos utilizados para la comprensión de texto, imágenes y secuencias de imágenes (vídeo). Además se vieron los futuros avances que se están desarrollando en ésta disciplina.

En el segundo bloque se vieron las diferentes técnicas que se utilizan en la actualidad para restaurar y reconstruir imágenes.

### 1.2.3. Optimización, heurística y multiobjetivo. Estrategias de paralelización

**Número de créditos:** 4.

**Profesorado:**

- Consolación Gil Montoya.
- María Dolores Gil Montoya.

**Programa del curso:**

Bloque I: Introducción a la optimización.

Bloque II: Técnicas heurísticas.

- Enfriamiento simulado.
- Búsqueda Tabú.
- Computación evolutiva.
- Ejemplo de optimización heurística. Partición de grafos.

Bloque III: Optimización multiobjetivo.

- Introducción a la optimización multiobjetivo.
- Formulación del problema. Óptimo de Pareto.

- Optimización evolutiva multiobjetivo. Técnicas.
- Comparación experimental de diferentes técnicas.

Bloque IV: Estrategias de Paralelización.

- Distribución de datos.
- El problema del desbalanceo.
- Estrategias de grano fino y grueso.

#### **Motivación de la elección del curso:**

En la vida siempre se enfrentan problemas en donde se deben tomar decisiones tratando de llegar a un objetivo. La optimización es una herramienta que facilita dicha tarea. Sin embargo, muchos problemas de la vida real implican la optimización simultánea de varios objetivos en conflicto. La optimización multiobjetivo permite afrontar el problema de diseño en multitud de disciplinas, dado que proporciona soluciones a problemas de optimización de varias funciones objetivo simultáneas que pueden estar en conflicto (por ejemplo objetivos económicos frente a objetivos relacionados con las prestaciones).

Los problemas de optimización aparecen en numerosas aplicaciones, tales como planificación y asignación de recursos a tareas (job-scheduling), cadenas de montaje industrial, diseño y control de redes de distribución (agua, gas, etc.), diseño y test de circuitos, etc. En su mayor parte se trata de problemas NP-completos y por consiguiente cualquier procedimiento general de resolución, al necesitar unos recursos que crecen exponencialmente con el tamaño del problema, presenta importantes dificultades de cara a su aplicación a casos realistas.

Por tanto, en muchos casos hay que desarrollar heurísticas específicas. Así en los últimos años se han propuesto procedimientos para optimización multiobjetivo que se incluyen dentro de la llamada computación evolutiva. Se trata de procedimientos de gran flexibilidad que permiten de manera relativamente sencilla establecer un compromiso entre la calidad de la solución obtenida y el tiempo empleado en hallarla. Dado que la naturaleza de estos algoritmos es paralela de forma inherente, este curso ofrece una visión de las estrategias más usadas para paralelizar estos algoritmos así como los problemas que aparecen en relación a balanceo de carga y distribución de soluciones.

#### **1.2.4. Algoritmos de optimización global. Estrategias paralelas.**

**Número de créditos:** 5.

**Profesorado:** Leocadio González Casado.

#### **Programa del curso:**

Bloque I: Optimización Global.

- Introducción.
- Modelado Matemático, casos.
- Condiciones de optimalidad en programación no lineal NLPL.
- Ventajas de la Optimización Global.
- Algoritmos de Optimización no Lineal.

Bloque II: Algoritmos Metaheurísticos. Paralelización.

- Introducción a los algoritmos de Optimización Global estocásticos.
- Algoritmos Heurísticos.
- Metaheurísticas.
- Aplicaciones de algoritmos de optimización metaheurísticos.
- Introducción a la Paralelización.
- Paralelización de Metaheurísticas.

Bloque III: Algoritmos Determinísticos. Paralelización.

- Introducción a los algoritmos determinísticos de optimización global.
- Ramificación y Acotación. El problema de diseño de mezclas.
- Aritmética de Intervalos y diferenciación automática.
- Problemas de corte con cero.
- Aceleración de algoritmos de ramificación y acotación para la búsqueda de mínimos.
- Paralelización de algoritmos determinísticos.

**Motivación de la elección del curso:**

Esta asignatura contiene un contenido enfocado a estudiar las técnicas y algoritmos existentes para la resolución de problemas de optimización global, así como mecanismos propios de algoritmos estocásticos de optimización global.

También pretende que el alumno aprenda a dominar y manejar herramientas usadas en optimización global determinista como son la aritmética de intervalos, diferenciación automática y dispositivos aceleradores de algoritmos de ramificación y acotación, así como extraer el paralelismo de los problemas computacionalmente costosos que se resuelven mediante estos algoritmos de optimización global.

Asimismo, se ofrecen problemas reales a los que se pueden aplicar los conocimientos adquiridos en la asignatura.

### 1.2.5. Problemas matriciales en sistemas multiprocesador

**Número de créditos:** 5.

**Profesorado:**

- Inmaculada García Fernández.
- Gracia Ester Martín Garzón.

**Programa del curso:**

Bloque I: Introducción al procesamiento paralelo.

Bloque II: El tratamiento de la irregularidad en un entorno monoprocesador. Modelos de almacenamiento y referencia para estructuras irregulares.

Bloque III: Análisis de algunos ejemplos de problemas irregulares.

Bloque IV: Modelos de distribución / partición de datos.

Bloque V: Técnicas de balanceo de carga computacional en sistemas multiprocesador.

Bloque VI: Efectos de la irregularidad en la gestión de la jerarquía de memoria.

Bloque VII: Computación matricial paralela: operaciones fundamentales.

Bloque VIII: Software para computación matricial: BLAS, LAPACK, BLACS, ScaLAPACK, PLAPACK.

**Motivación de la elección del curso:**

En este curso se tomó contacto con el tratamiento de problemas usando multiprocesadores o sistemas de computación distribuida, y se ofrecieron los conocimientos necesarios para caracterizar el conjunto de problemas que pueden ser tratados por computadores paralelos, analizar las posibles causas de irregularidad y su tratamiento en arquitecturas de memoria distribuida y abordar problemas matriciales intentando obtener su solución paralela.

### 1.2.6. Balanceo dinámico de carga en cluster computing. Aplicación en Tomografía Computerizada

**Número de créditos:** 3.

**Profesorado:**

– Javier Roca Piera.

**Programa del curso:**

Bloque I: Arquitecturas paralelas escalables. Paso de mensajes (MPI). Rendimiento.

Bloque II: Balance de carga. Balance de carga estático y dinámico. Estrategias.

Bloque III: Aplicación. Reconstrucción tomográfica 3D desde imágenes.

– Algoritmos de reconstrucción.

– Paralelismo usando paso de mensajes.

Bloque IV: Entornos multithread. Memoria compartida y paralelismo de escritorio: POSIX thread.

**Motivación de la elección del curso:**

Desde que en 1980 se iniciara el reto del procesamiento paralelo para mejorar la eficiencia en la resolución de problemas computacionalmente costosos, muchas han sido las arquitecturas que se han desarrollado. En los últimos años la conexión mediante distintas tecnologías de red, de agrupaciones de procesadores multicore, ha sido la configuración más utilizada como sistema multicomputador. Nuevas soluciones se han tenido que aportar a problemas ya existentes, como por ejemplo el balanceo dinámico de carga. Independientemente de que el cluster de máquinas computadoras sea dedicado o no, podemos enunciar una de las necesidades a satisfacer en estas arquitecturas: dado un conjunto de tareas y de datos a procesar y un conjunto de procesadores disponibles, encontrar una asignación de tareas y datos a los procesadores que consiga que cada procesador tenga aproximadamente la misma cantidad de trabajo.

En este curso se analizan distintas estrategias de balanceo dinámico de carga, valorando los parámetros que intervienen en su rendimiento. Como aplicación sobre la que comprobar de forma práctica estas estrategias se utiliza la reconstrucción tridimensional de imágenes a partir de proyecciones.



### 1.3. RESUMEN Y CONCLUSIONES DEL CAPÍTULO

Durante el período de docencia he completado un número de créditos mucho mayor del necesario, debido a que he visto necesario cursar bastantes cursos de doctorado con el objetivo de obtener la máxima formación académica posible para fortalecer mi futuro proceso de investigación.

Como se comentó en apartados anteriores, los cursos de doctorado los he expuesto en este capítulo en orden de relevancia para mi línea de investigación principal: clasificación de imágenes de satélite mediante autómatas celulares. De este modo, he situado en primer lugar el curso de doctorado *modelado de sistemas de información* porque mi trabajo de investigación está incluido claramente en este área. A continuación, el curso *metodologías de compresión, restauración y reconstrucción de imagen* ha fortalecido mis conocimientos sobre los formatos existentes de imágenes así como su tratamiento digital, un aspecto también importante para mi tema de investigación. El resto de cursos de doctorado, todos relacionados con procesos de paralelización y optimización algorítmica, tienen un contenido teórico y práctico que podré utilizar en el futuro en las mejoras algorítmicas del trabajo que expongo en el presente documento.



---

## CAPÍTULO 2

# TELEDETECCIÓN

---



# Capítulo 2

## TELEDETECCIÓN

### Contenidos

---

<b>2.1.</b>	<b>INTRODUCCIÓN A LA TELEDETECCIÓN . . . . .</b>	<b>19</b>
2.1.1.	Definición de teledetección . . . . .	19
2.1.2.	Historia de la teledetección . . . . .	20
2.1.3.	Elementos básicos de la teledetección . . . . .	22
2.1.4.	Aspectos físicos de la teledetección . . . . .	23
<b>2.2.</b>	<b>CARACTERÍSTICAS DE LAS IMÁGENES DE SATÉLITE . . . . .</b>	<b>24</b>
2.2.1.	Formato de las imágenes de satélite . . . . .	24
2.2.2.	Resolución de las imágenes de satélite . . . . .	28
2.2.3.	Errores en la captación de imágenes . . . . .	29
2.2.4.	Clasificación de imágenes de satélite . . . . .	29
<b>2.3.</b>	<b>ALGORITMOS DE CLASIFICACIÓN NO SUPERVISADOS . . . . .</b>	<b>32</b>
2.3.1.	Clasificador k-medias . . . . .	32
2.3.2.	Clasificador isodata . . . . .	33
<b>2.4.</b>	<b>ALGORITMOS DE CLASIFICACIÓN SUPERVISADOS . . . . .</b>	<b>35</b>
2.4.1.	Clasificador mínima distancia . . . . .	35
2.4.2.	Clasificador paralelepípedos . . . . .	36
2.4.3.	Clasificador máxima verosimilitud . . . . .	37
<b>2.5.</b>	<b>RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .</b>	<b>38</b>

---



**A**ctualmente existen en órbita decenas de satélites artificiales de observación de la Tierra que generan miles de imágenes cada día para todo tipo de aplicaciones de uso civil y militar. Debido a la enorme cantidad de información que se genera diariamente relacionada con la teledetección, es necesario la ayuda de algoritmos computacionales que automaticen los procesos de análisis e interpretación de las imágenes de satélite, y por este motivo una gran cantidad de grupos investigación de todo el mundo están trabajando en la actualidad en el desarrollo de nuevas técnicas informáticas que optimicen los resultados obtenidos por los algoritmos computacionales ya existentes en el ámbito de la teledetección. Este capítulo ofrece una introducción a la ciencia aplicada llamada teledetección, y está dividido en los siguientes apartados principales:

- Introducción a la teledetección: en este apartado se define el término teledetección, se ofrece un resumen de la historia de la teledetección, se enumeran los elementos básicos que intervienen en un sistema genérico de teledetección y se explican los aspectos físicos en los que se basa esta importante ciencia aplicada.
- Características de las imágenes de satélite: en este apartado se enumeran los formatos más comunes que se utilizan para representar las imágenes de satélite, se diferencian los distintos tipos de resolución que existen, se explican los errores más comunes que pueden ocurrir durante el proceso de captación de las imágenes de satélite y se define una de las técnicas más importantes para su análisis e interpretación: los algoritmos de clasificación de imágenes de satélite.
- Algoritmos de clasificación no supervisados: en este apartado se explica el funcionamiento de dos algoritmos de clasificación de imágenes de satélite no supervisados, los clasificadores k-medias e isodata.
- Algoritmos de clasificación supervisados: en este apartado se explica el funcionamiento de tres algoritmos de clasificación de imágenes de satélite supervisados, los clasificadores mínima distancia, paralelepípedos y máxima verosimilitud.
- Resumen y conclusiones del capítulo: en este apartado se destacan los aspectos de la teledetección en los que se ha centrado mi trabajo de investigación.

## 2.1. INTRODUCCIÓN A LA TELEDETECCIÓN

En este apartado se define el término teledetección, se ofrece un resumen de la historia de la teledetección, se enumeran los elementos básicos que intervienen en un sistema genérico de teledetección y se explican los aspectos físicos en los que se basa esta importante ciencia aplicada.

### 2.1.1. Definición de teledetección

Las Naciones Unidas definen el término teledetección como “la observación de la superficie terrestre desde el espacio, utilizando las propiedades de las ondas electromagnéticas emitidas, reflejadas o difractadas por los objetos observados, para fines de mejoramiento de la ordenación de los recursos naturales, de utilización de tierras y de protección del medio ambiente” [2].

Desde un punto de vista estrictamente técnico, podemos definir la teledetección como “el conjunto de técnicas, aparatos y procedimientos que permiten obtener y analizar imágenes de la superficie de la Tierra desde sensores ubicados remotamente” [15]. La palabra teledetección realmente corresponde a la traducción de la expresión inglesa *remote sensing*, ciencia aplicada que surgió a principio de los 60's para definir los métodos de observación remota de la superficie de la Tierra. Aunque en sus orígenes este término se aplicó principalmente a la fotografía aérea, posteriormente también se incluyó la observación a través de plataformas de observación satelitales, surgiendo así la teledetección satelital.

La teledetección es una ciencia importante que nos permite la adquisición de información de la superficie terrestre y de valores ambientales sin necesidad de tener contacto real con el área que está siendo observada [47]. Esta ciencia se puede utilizar en muchas aplicaciones medioambientales, ayudando a resolver y mejorar los problemas sociales derivados de ellos. Entre el amplio repertorio de aplicaciones de la teledetección podemos encontrar ejemplos que incluyen estudios de calidad del suelo, búsqueda de recursos hídricos, protección del medio ambiente o simulaciones meteorológicas, entre otros.

La teledetección se ha convertido en la actualidad en una herramienta indispensable para la humanidad que puede ser usada en numerosos aspectos de enorme relevancia medioambiental, militar, social, económica y política, como analizar y predecir fenómenos meteorológicos, realizar espionaje militar para obtener ventaja estratégica, prevenir desastres naturales y evitar pérdidas humanas, mejorar las técnicas de producción agrícola y ganadera, y registrar la ocupación y uso del suelo de la superficie terrestre, respectivamente.

### 2.1.2. Historia de la teledetección

El origen de la teledetección se remonta a mediados del siglo XIX y abarca desde las primeras fotografías aéreas tomadas con globo, pasando por las fotografías aéreas tomadas desde un avión, hasta las avanzadas imágenes digitales captadas por los sensores ubicados en sofisticados satélites espaciales. Barret y Curtis [9] dividen la historia de la teledetección en siete períodos distintos:

- *Primer período: inicio-1925.* Este período, el más largo de todos, abarca desde la captación de la primeras fotografías aéreas a mediados del siglo XIX hasta la finalización de la primera guerra mundial. La historia de la teledetección tiene sus orígenes en 1839, año en el que se captaron las primeras fotografías aéreas en Francia, para posteriormente realizar las primeras aplicaciones al mapeo topográfico concebidas por el director del Observatorio de París. Unos años más tarde, en 1858, Gaspard Felix Tournachon mapeó la ciudad de París ascendiendo sobre un globo.

Gracias a los avances tecnológicos de la fotografía que se produjeron hacia 1871, se pudo construir cámaras mucho más livianas, lo que impulsó el uso del globo para la captación de fotografías aéreas. En 1906 G. Lawrence captó los daños ocasionados por el terremoto de San Francisco desde una altitud de 600 metros. Las primeras fotografías aéreas tomadas desde un avión las captó Wilbur Wright en 1909 sobre Centocelli, Italia.

Durante la primera guerra mundial se utilizó la fotografía aérea captada desde avión como herramienta de inteligencia con el objetivo de conseguir ventajas estratégicas en el campo de batalla. Este acontecimiento supuso un punto de inflexión en la historia de la teledetección, ya que a partir de este momento, demostrada su eficacia en aplicaciones militares, se comenzaron a invertir muchos más recursos económicos.



Durante el primer período de la historia de la teledetección quedó demostrado la utilidad de fotografías aéreas, obtenidas en globos aerostáticos o en aviones, para conseguir ventaja estratégica militar.

- *Segundo período: 1925-45.* Esta fase abarca desde la finalización de la primera guerra mundial hasta la finalización de la segunda guerra mundial. La segunda guerra mundial dio un gran impulso a la tecnología de las fotografías aéreas, en el desarrollo de nuevas cámaras y técnicas de reconocimiento, lo que permitió mejorar la interpretación de la superficie terrestre y aumentar más aún la ventaja estratégica militar.
- *Tercer período: 1945-55.* Esta etapa se caracterizó por una gran difusión y diversificación de las técnicas de fotografía aérea, que comenzaron a aplicarse a diversas disciplinas no militares: agricultura, geología, arqueología, etc.
- *Cuarto período: 1955-60.* Durante esta fase se produjeron los primeros lanzamientos de satélites artificiales. El 4 de octubre de 1957 la URSS lanzó con éxito el Sputnik I, primer satélite artificial de la historia. Posteriormente se lanzó el Sputnik II, satélite que puso en órbita al primer ser vivo en el espacio: la perrita Laika.
- *Quinto período: 1960-80.* Durante este período surgió la teledetección satelital, que posibilitó la adquisición sistemática de todo tipo de información de la superficie de la Tierra sin tener contacto físico con la zona de estudio concreta. A lo largo de este documento, cada vez que se cite el término teledetección, se hará referencia a la teledetección satelital. Los programas espaciales permitieron impulsar los avances tecnológicos relacionados con las técnicas de teledetección. En 1960 se lanzó el satélite TIROS-I, primer satélite meteorológico de una familia de satélites posteriormente rebautizada como NOAA, que permitía distinguir entre nubes, agua, hielo y nieve. En 1969 el vuelo tripulado del Apolo 9 recogió las primeras fotografías orbitales multiespectrales para el estudio de los recursos terrestres. En 1972 se puso en órbita el primer satélite dedicado específicamente a la teledetección: el LANDSAT 1, primero de la familia de satélites LANDSAT puestos en órbita por la NASA.
- *Sexto período: 1980-95.* Esta fase se caracteriza por el gran desarrollo de software para la interpretación computerizada de las imágenes de satélite digitales. Este proceso de automatización comenzó a ser necesario como consecuencia del aumento de la cantidad de datos obtenidos a través de las cada vez más numerosas plataformas satelitales. La puesta en marcha durante esta fase del transbordador espacial y de la estación espacial MIR influyeron positivamente en el avance de las técnicas de teledetección espacial. Durante esta etapa, además, se lanzaron varios satélites muy importantes en el ámbito de la teledetección, como el satélite francés SPOT en 1985, el IRS de la India en 1988 y el ERS de la Agencia Espacial Europea en 1991.
- *Séptimo período: 1995-actualidad.* Los últimos años de la historia de la teledetección se han caracterizado sobre todo por la comercialización cada vez más creciente de la información obtenida mediante técnicas de teledetección satelital. Esto ha provocado que, información que en otros períodos de la historia de la teledetección hubiera sido clasificada como de espionaje militar debido a su alta relevancia estratégica, esté disponible de manera gratuita para cualquier ser humano del planeta con acceso a Internet (ej: Google Maps).

Para evitar futuras apropiaciones políticas del espacio exterior existe un foro llamado COPUOS (Committee on the Peaceful Uses of Outer Space) que se encarga del desarrollo de normativa internacional del espacio, con el objetivo de que siempre sea patrimonio común de la Humanidad desde un punto de vista formalmente jurídico. Para ello se han establecido varios tratados y acuerdos internacionales que reglamentan todas las actividades relacionadas con el espacio exterior.

### 2.1.3. Elementos básicos de la teledetección

Un sistema genérico de teledetección suele incluir los siguientes elementos [15], que se muestran en la Figura 2.1:

- *Fuente de energía*, que produce la radiación electromagnética captada por los sensores del satélite, que puede ser pasiva (como la luz solar) o activa (emitida por el propio sensor para posteriormente captar el reflejo).
- *Superficie terrestre*, natural o artificial, que refleja la radiación electromagnética.
- *Sistema sensor*, formado por los sensores (cámaras, radar, etc) y la plataforma de observación (satélite, avión, etc), que recoge la radiación electromagnética emitida por la superficie terrestre y la envía al sistema de recepción.
- *Sistema de recepción*, que recibe y almacena la información del sistema sensor.
- *Intérprete*, que transforma los datos recibidos en información útil para el usuario.
- *Usuario*, que utiliza la información interpretada con algún objetivo concreto.

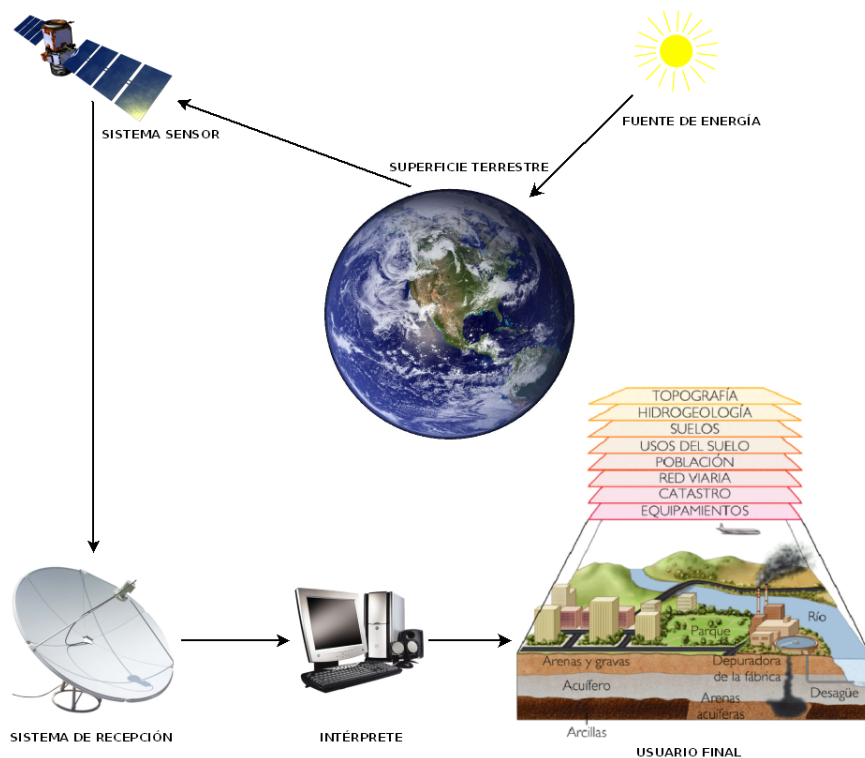


Figura 2.1: Elementos básicos de un sistema genérico de teledetección.

### 2.1.4. Aspectos físicos de la teledetección

Según la termodinámica, importante rama de la física, existen tres mecanismos fundamentales de transmisión de energía térmica: conducción, convección y radiación, siendo este último mecanismo el único en el que se transmite energía sin contacto físico entre el emisor y el receptor. Existen muchos tipos de radiación: radiación electromagnética, radiación térmica, radiación nuclear, radiación ionizante, etc. De todos los tipos de radiación existentes, en este apartado nos vamos a centrar en la radiación electromagnética, ya que es una de las bases físicas más importantes de la teledetección [25]. La naturaleza de la radiación electromagnética se puede explicar a partir del modelo de onda y el de partícula.

El *modelo de onda*, desarrollado por científicos como Huygens y Maxwell, establece que la radiación se produce como resultado de oscilaciones en los campos eléctrico y magnético de los objetos, lo que genera ondas con dos campos de fuerzas ortogonales entre sí (el eléctrico y el magnético), que transmiten energía. Los dos parámetros básicos que definen las ondas son la longitud y la frecuencia. Son inversamente proporcionales y se relacionan entre sí a través de la siguiente fórmula matemática:

$$\lambda = \frac{v}{f} \quad (2.1)$$

donde  $\lambda$  es la longitud de onda,  $f$  es la frecuencia y  $v$  la velocidad.

El *modelo de partícula*, desarrollado por científicos como Planck y Einstein, establece que la energía se transmite como un flujo de partículas llamadas fotones. Los modelos de onda y partícula se relacionan entre sí a través de la siguiente fórmula:

$$E = \frac{hc}{\lambda} \quad (2.2)$$

donde  $E$  es la energía de los fotones,  $h$  es la constante de Planck ( $6,626 \times 10^{-34}$  Js),  $c$  es la velocidad de la luz ( $3 \times 10^8$  m/s) y  $\lambda$  es la longitud de onda.

El espectro electromagnético es la distribución energética de las longitudes de onda que puede adoptar la radiación electromagnética. Se extiende desde las ondas electromagnéticas de menor longitud de onda, como los rayos gamma y los rayos X, pasando por la luz ultravioleta, luz visible y rayos infrarrojos, hasta las radiaciones de mayor longitud de onda, como las ondas de radio. La Figura 2.2 muestra las franjas del espectro electromagnético más usadas por las técnicas de teledetección.

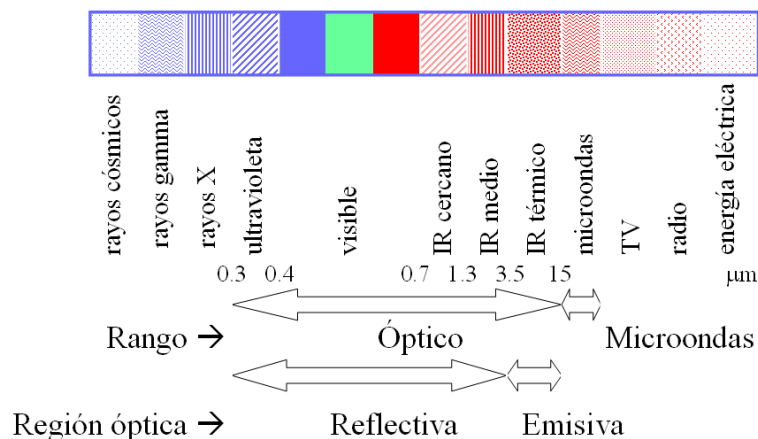


Figura 2.2: Espectro electromagnético y técnicas de teledetección.

## 2.2. CARACTERÍSTICAS DE LAS IMÁGENES DE SATÉLITE

En este apartado se enumeran los formatos más comunes que se utilizan para representar las imágenes de satélite, se diferencian los diferentes tipos de resolución que existen para este tipo de imagen, se explican los errores más comunes que pueden ocurrir durante el proceso de captación de las imágenes de satélite y se define una de las técnicas más importantes para su análisis e interpretación: los algoritmos de clasificación de imágenes de satélite.

### 2.2.1. Formato de las imágenes de satélite

Una imagen de satélite es un archivo tipo raster formado por una matriz de celdas, donde cada celda se denomina píxel. A cada píxel se le asignan varios valores digitales, que corresponden a la reflectividad recogida por un sensor específico. Las imágenes de satélite son almacenadas en diversas capas o bandas, donde cada banda almacena el valor que corresponde a cada píxel de la imagen en un intervalo concreto del espectro electromagnético. Por lo tanto una imagen de satélite es en realidad un conjunto de imágenes, con las mismas propiedades geométricas, donde cada imagen almacena el valor de reflectancia de los píxeles en un intervalo de longitud de onda concreto del espectro electromagnético.

Las imágenes de satélite pueden tener un número reducido de bandas, como es el caso de las imágenes multiespectrales (menos de 10 bandas) o un número muy elevado, como ocurre con las imágenes hiperespectrales (con cientos de bandas). La Figura 2.3 muestra una imagen multiespectral y una imagen hiperespectral.

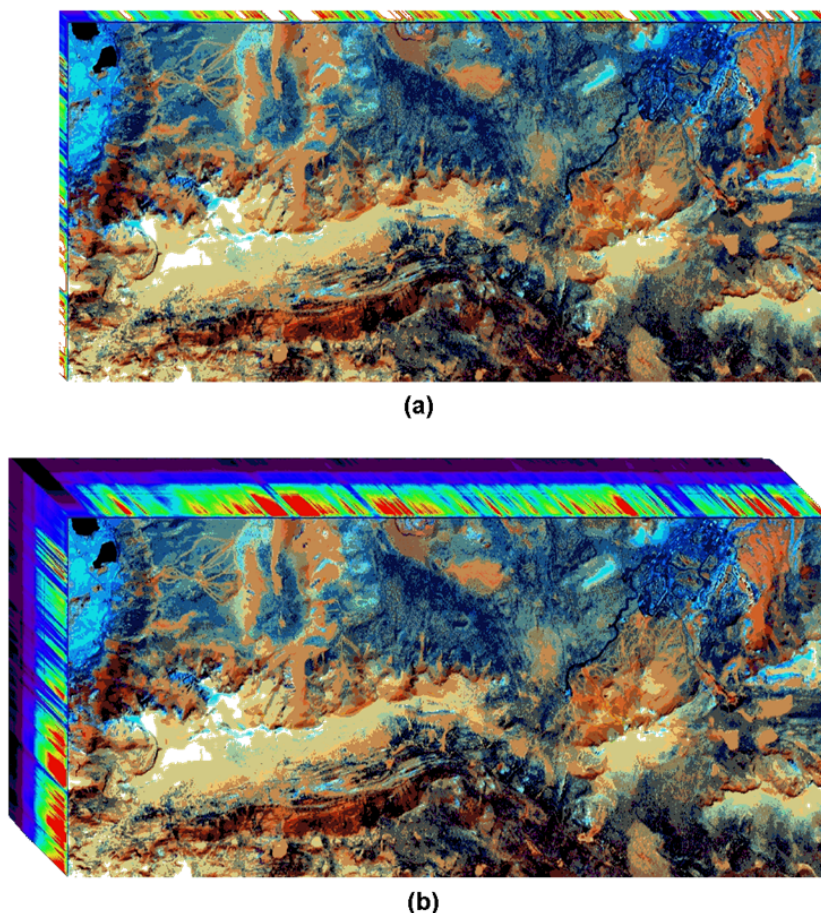


Figura 2.3: (a) Imagen multiespectral. (b) Imagen hiperespectral.

Las imágenes de satélite también se pueden representar como modelos de elevación digital o DEM (Digital Elevation Model), donde cada píxel almacena la altura a la que se encuentra la superficie terrestre en ese punto determinado de la imagen. De este modo, se pueden realizar representaciones tridimensionales donde se muestra la elevación del terreno de la imagen. La Figura 2.4 muestra un ejemplo de modelo de elevación digital (DEM).

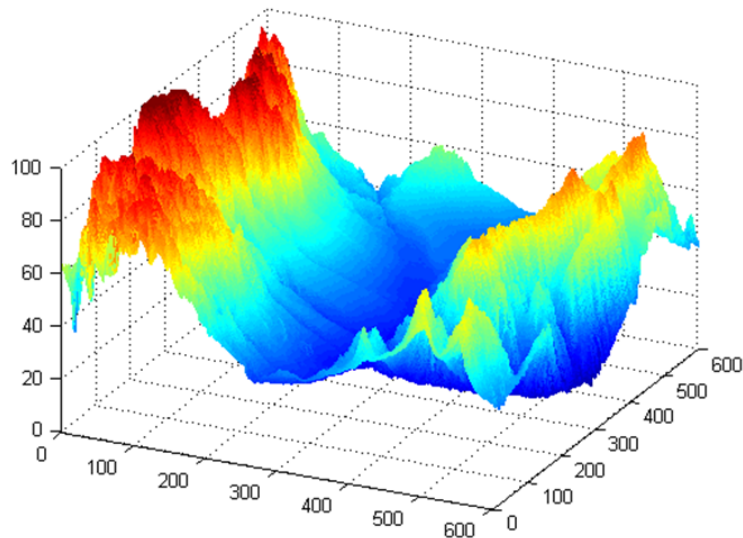


Figura 2.4: Imagen DEM (Digital Elevation Model).

Las imágenes de satélite también se pueden representar desde el punto de vista espectral en un espacio de características, que es una gráfica no espacial donde se representan las clases con sus características principales (centro, variabilidad, etc) y los píxeles como los puntos correspondientes a sus valores espectrales. La Figura 2.5 muestra un ejemplo de representación de una imagen de satélite en el espacio de características.

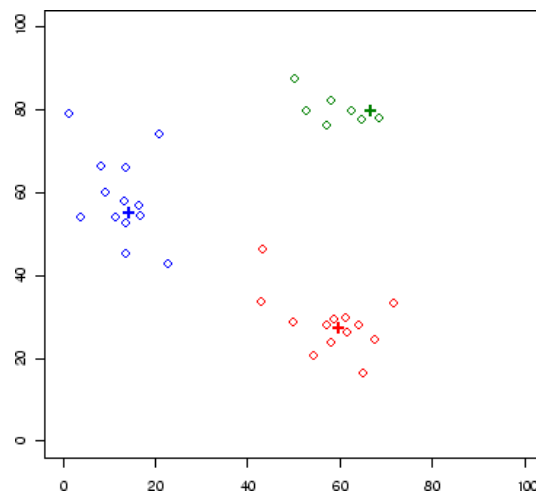
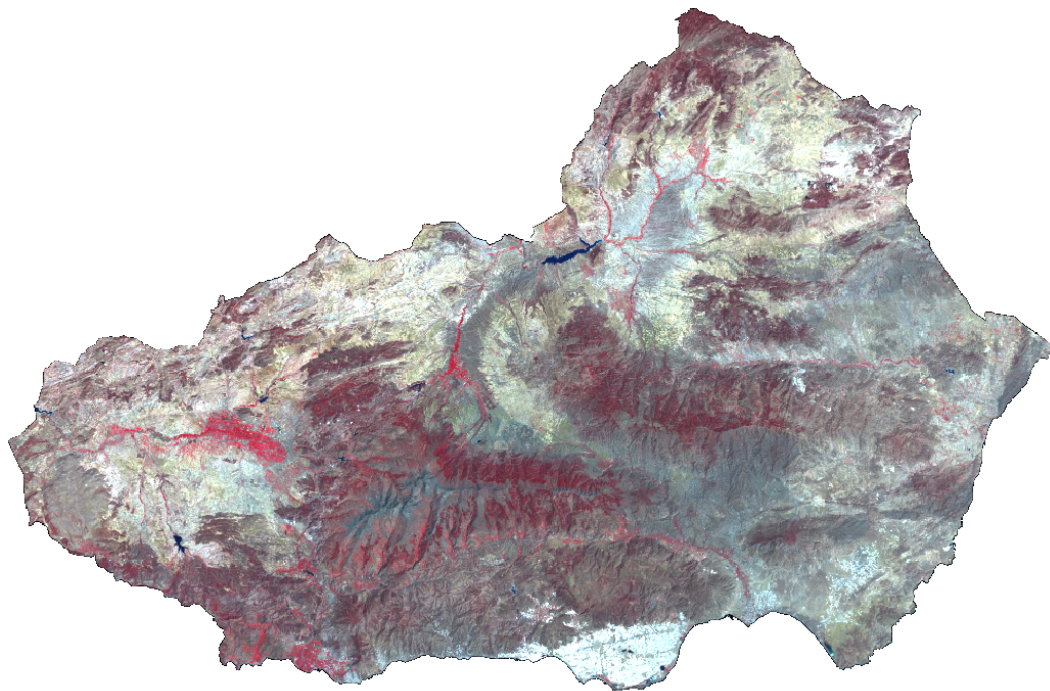


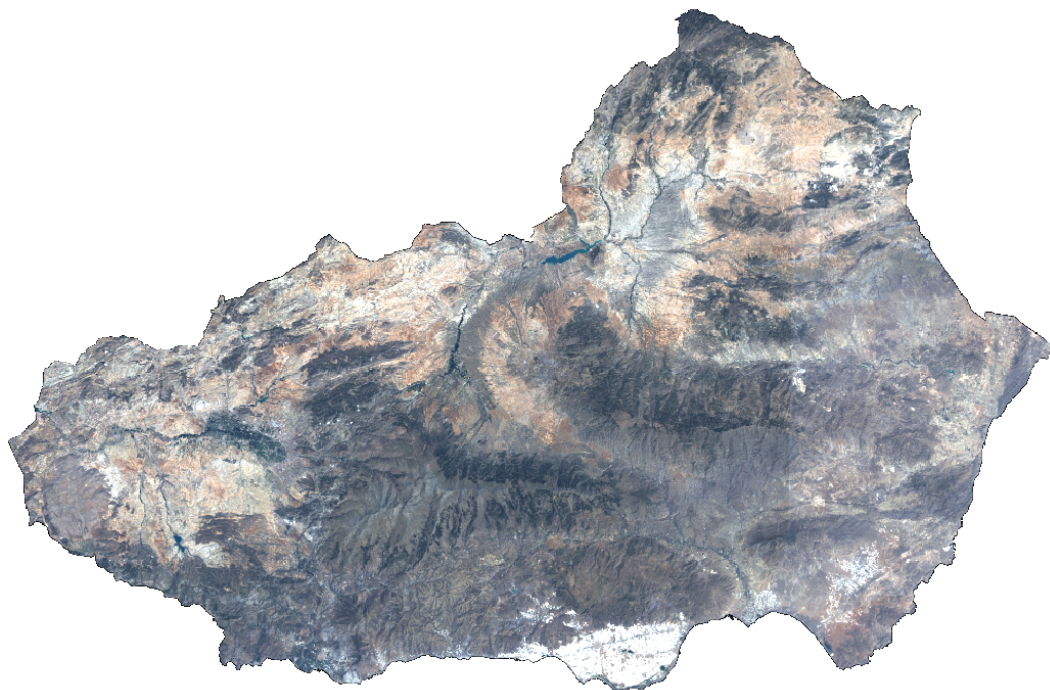
Figura 2.5: Imagen de satélite en el espacio de características (3 clases distintas).

La Figura 2.6 muestra imágenes de satélite de las provincias Almería y Granada (sureste de España), combinando diversas bandas para conseguir distintos resultados. La Figura 2.7 muestra diversos fenómenos naturales atmosféricos captados mediante satélite.





(a)



(b)

Figura 2.6: (a) Imagen Landsat-7 de las provincias Almería y Granada (bandas 2, 3 y 4). (b) Imagen Landsat-7 de las provincias Almería y Granada (bandas 1, 2 y 3: color verdadero).



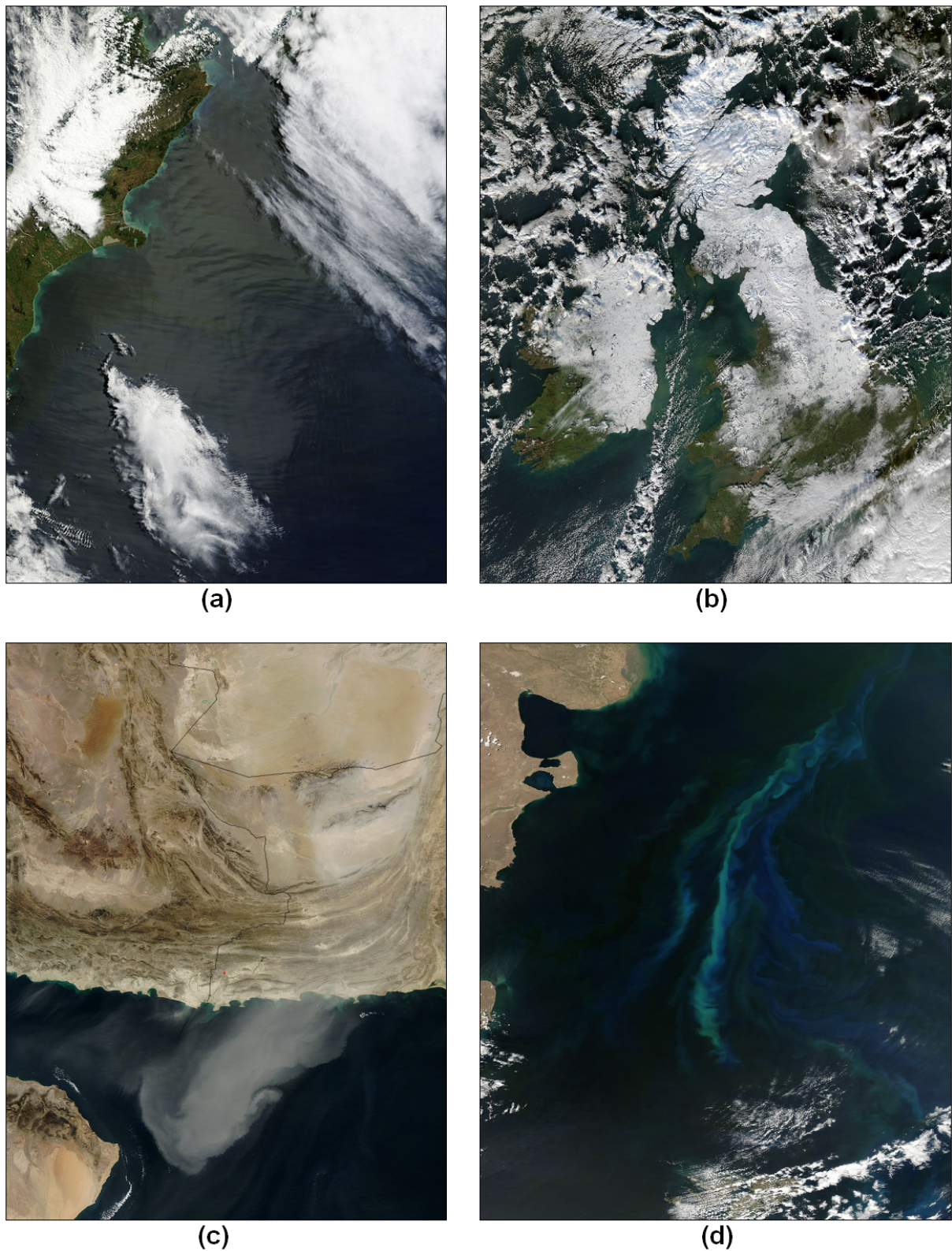


Figura 2.7: (a) Olas de gravedad atmosférica en Nueva Zelanda. (b) Nieve en Gran Bretaña e Irlanda. (c) Tormentas de polvo en el suroeste de Asia. (d) Floración de fitoplancton en Argentina. (Fuente: NASA).

### 2.2.2. Resolución de las imágenes de satélite

Los satélites captan la radiación emitida o reflejada por la superficie terrestre en 4 dimensiones: espacio, tiempo, longitud de onda y radiancia. Toda esta información continua debe ser muestreada por los sensores para convertirla en datos digitales discretos. Este proceso de discretización tiene por lo tanto cuatro resoluciones posibles:

- *Resolución espacial*: es la superficie terrestre que almacena cada píxel de la imagen. También se suele usar el concepto de *IFOV*, o campo instantáneo de visión, que se define como la sección angular en radianes observada en un momento determinado. La siguiente ecuación muestra la relación entre *IFOV* y el tamaño de píxel:

$$d = 2H \tan\left(\frac{IFOV}{2}\right) \quad (2.3)$$

donde  $d$  es el tamaño de píxel y  $H$  la distancia del sensor a la superficie terrestre.

Este valor puede variar bastante dependiendo del tipo de satélite que tome la imagen: desde el satélite Meteosat destinado a captar información meteorológica, que ofrece una resolución espacial de 5000 m, pasando por los satélites Lansat-TM que poseen una resolución espacial de 30 m, hasta el sensor Ikonos-1 que permite una resolución de 1 m. Dependiendo del tipo de fenómeno que queramos estudiar, convendrá elegir un tipo de resolución u otra. Si el estudio está enfocado a grandes fenómenos meteorológicos, la resolución que ofrece el satélite Meteosat será adecuado para poder disponer de una cobertura global de la superficie terrestre. Si deseamos estudiar la cobertura ecológica de una determinada zona, la resolución espacial ofrecida por el satélite Lansat-TM puede ser adecuada, debido a que no es ni muy elevada ni demasiado reducida. En cambio, si queremos realizar un estudio de objetos de manera individual, necesitamos utilizar la resolución ofrecida por el sensor Ikonos-1. Por lo tanto, la resolución espacial que elijamos para nuestro estudio debe ser lo suficientemente grande como para poder disponer de la información que necesitamos, y lo suficientemente pequeña como para descartar información redundante. Mantener dicho equilibrio puede ahorrarnos costes computacionales innecesarios, así como resultados erróneos por no disponer de suficiente información.

- *Resolución temporal*: es el intervalo de tiempo que transcurre entre cada imagen obtenida por el sensor en la misma zona de la superficie terrestre. Esta resolución puede ser la que queramos en el caso de los aviones, cada media hora en el caso de los satélites geosíncronos y variable en el caso de los satélites heliosíncronos.
- *Resolución espectral*: es el número y anchura de las bandas electromagnéticas captadas por el sensor. A mayor número de bandas, dispondremos de mayor número de variables que puedan describir cada píxel de la imagen. Por otro lado, las bandas estrechas aumentan el poder discriminante de los valores captados por el sensor sobre las bandas anchas. El número de bandas y la anchura depende del objetivo que se pretende cubrir con la información captada por el sensor.
- *Resolución radiométrica*: es la sensibilidad de discriminar entre pequeñas variaciones en la radiación captada por el sensor. Esta resolución se suele expresar como el número de bits necesarios para almacenar cada píxel, lo que nos ofrece el número de niveles digitales (*ND*) del sensor. A mayor resolución radiométrica, mayor información será captada por el sensor.



### 2.2.3. Errores en la captación de imágenes

Existen diversos factores que pueden influir negativamente en el proceso de captación de la radiación electromagnética de la superficie terrestre por parte de los sensores de las plataformas satelitales, introduciendo errores no deseados en los datos recogidos. Estos errores se pueden clasificar en tres grandes categorías: geométricos, radiométricos y atmosféricos. A continuación se muestran los errores más frecuentes dentro de cada categoría:

- *Errores geométricos*: provocados por la propia plataforma satelital, por los sensores instalados en ella o por el fenómeno de rotación terrestre. Las distorsiones provocadas por la plataforma satelital tienen su origen en oscilaciones aleatorias de su altitud, orientación y velocidad, que alteran de manera impredecible la relación que se establece entre la superficie terrestre y las posiciones de la imagen adquirida. Los errores introducidos por los sensores se deben a la elevada complejidad del proceso de captación de la imagen, y puede producir que no todos los píxeles de la imagen tengan la misma resolución. Por último, el fenómeno natural de rotación de la Tierra produce que la superficie terrestre se desplace espacialmente desde el momento de inicio del proceso de la captura de la imagen hasta el final del mismo, debido a que dicho proceso puede llegar a requerir bastante tiempo. Todos estos errores se pueden solucionar gracias a la corrección geométrica, mediante un proceso llamado georreferenciación de la imagen.
- *Errores radiométricos*: provocados por la descalibración de algún detector que los sensores poseen por cada una de las bandas electromagnéticas que captan, lo que puede producir efectos de bandeo en la imagen obtenida. En algunos casos pueden perderse algunos píxeles o líneas enteras. Estos errores pueden solucionarse mediante la corrección radiométrica.
- *Errores atmosféricos*: provocados por la interacción que se produce entre la radiación electromagnética y la atmósfera. Estos errores pueden solucionarse mediante la corrección atmosférica.

### 2.2.4. Clasificación de imágenes de satélite

La clasificación de imágenes de satélite es un proceso que consiste en agrupar los píxeles en un número finito de clases, basándose en los valores espectrales de las distintas bandas, convirtiendo de este modo la información captada por los sensores del satélite como niveles digitales a una escala categórica fácil de interpretar por los expertos analistas [39]. Los píxeles que pertenezcan a la misma clase deberán tener unas características similares [49].

Los algoritmos de clasificación de imágenes de satélite son una de las técnicas más importantes utilizadas en el ámbito de la teledetección, ya que facilitan a los investigadores la interpretación de la gran cantidad de información contenida en sus bandas. El objetivo de los algoritmos de clasificación de imágenes de satélite consiste en dividir los píxeles de la imagen en distintas clases, llamadas clases espectrales, teniendo en cuenta la similitud existente entre dichos píxeles. La imagen que se obtiene como resultado consiste básicamente en un mapa temático de la imagen original [49]. Los analistas utilizan posteriormente la imagen clasificada para interpretar de manera más sencilla la información contenida en la imagen de satélite. Estos algoritmos han alcanzado un gran avance en los últimos años y en la literatura existen diferentes procedimientos para clasificar imágenes satelitales.

La Figura 2.8 muestra los pasos del proceso genérico de clasificación de una imagen de satélite.

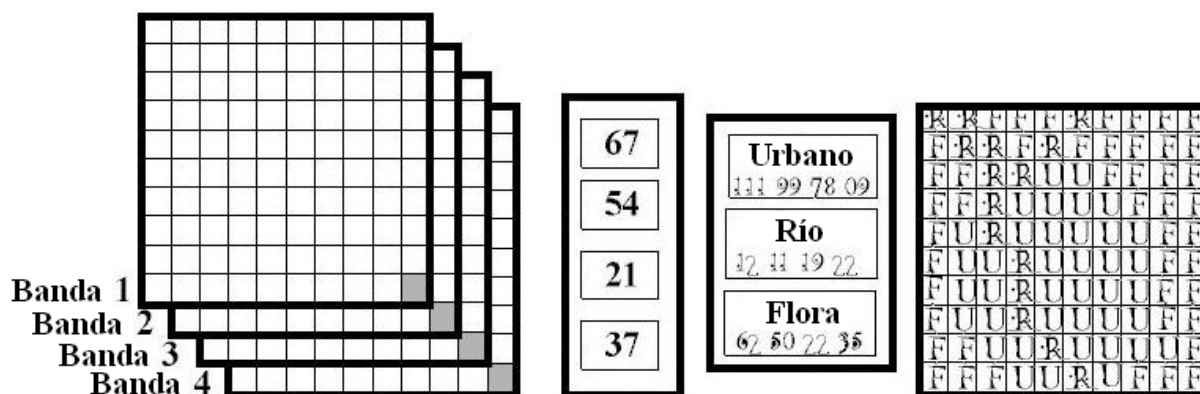


Figura 2.8: Proceso general de clasificación de imágenes de satélite.

En cualquier algoritmo de clasificación de imágenes podemos encontrar las siguientes fases genéricas:

- *Fase de entrenamiento*: definición de las clases en las que se van a agrupar los píxeles de la imagen.
- *Fase de asignación*: agrupación de los píxeles en las clases previamente definidas.
- *Verificación*: comprobación de los resultados obtenidos.

Los algoritmos de clasificación de imágenes de satélite se pueden dividir en dos grandes grupos basados en el método utilizado durante el procedimiento: clasificación supervisada y clasificación no supervisada, ambos tipos basados en las propiedades espectrales de los píxeles de la imagen. La utilización de procedimientos supervisados o no supervisados depende fundamentalmente del conocimiento experto que posee el analista sobre la zona de estudio [4]. La Figura 2.9 muestra el proceso general de clasificación supervisada y no supervisada, cada uno en un diagrama.

En los algoritmos de clasificación no supervisada, el analista sólo especifica al algoritmo el número de clases en las que desea dividir la imagen, y el propio algoritmo se encarga de agrupar píxeles similares basándose únicamente en la información espectral almacenada en las bandas de cada píxel. En este tipo de algoritmos, por tanto, no es necesario que el analista conozca la zona a estudiar. Existen muchos algoritmos de clasificación no supervisada. Los algoritmos clasificadores k-medias e isodata son dos de los procedimientos más utilizados para la clasificación no supervisada de imágenes de satélite.

En una clasificación supervisada, el analista selecciona muestras de píxeles de cada clase (conjunto de entrenamiento) para que el algoritmo pueda realizar una clasificación más precisa, basada en el conocimiento humano experto. En este tipo de algoritmos, el conocimiento que posee el analista sobre el área de estudio determina la calidad del conjunto de entrenamiento, y por lo tanto es un factor muy influyente. Los píxeles son etiquetados con la clase a la que más se asemejen a nivel espectral [5]. Existen muchos algoritmos de clasificación supervisada, y algunos de los más populares son la clasificación con mínima distancia, paralelepípedos y máxima verosimilitud.

Tanto los algoritmos de clasificación espectral supervisados como los no supervisados funcionan bastante bien en imágenes que no están alteradas por el ruido y donde las propiedades espectrales de los píxeles pueden establecer las clases con suficiente certeza, sin que existan clases muy próximas entre sí.

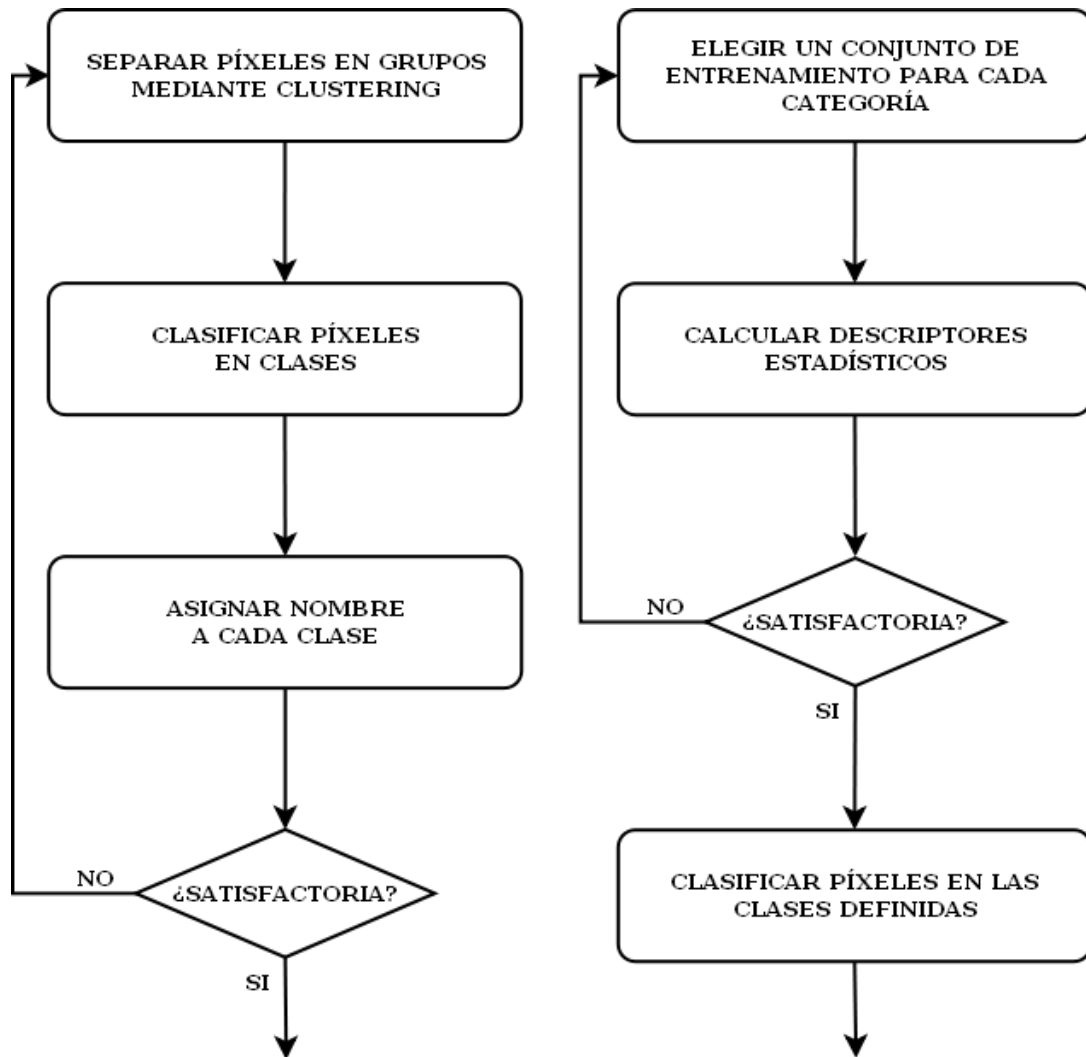


Figura 2.9: (Izquierda) Clasificación no supervisada. (Derecha) Clasificación supervisada.

Sin embargo, si las imágenes se encuentran alteradas por ruido de tipo impulsivo gaussiano o existen variaciones sustanciales en las propiedades de los píxeles, la imagen resultante puede tener muchas pequeñas regiones (a menudo de un píxel) que están mal clasificadas.

Para solucionar este error de clasificación se pueden utilizar técnicas adicionales a las espectrales, como el uso de información contextual, es decir, tener en cuenta para el proceso de clasificación de un píxel no sólo sus valores espectrales, sino además los de los píxeles de alrededor. Hay varios algoritmos de clasificación contextual que utilizan valores promedios, valores máximos o descripción de las texturas para mejorar la clasificación espectral.

La Figura 2.10 muestra la descripción del proceso general de clasificación (a), así como los resultados obtenidos en una clasificación supervisada (b) y una clasificación no supervisada (c) de una imagen de satélite.

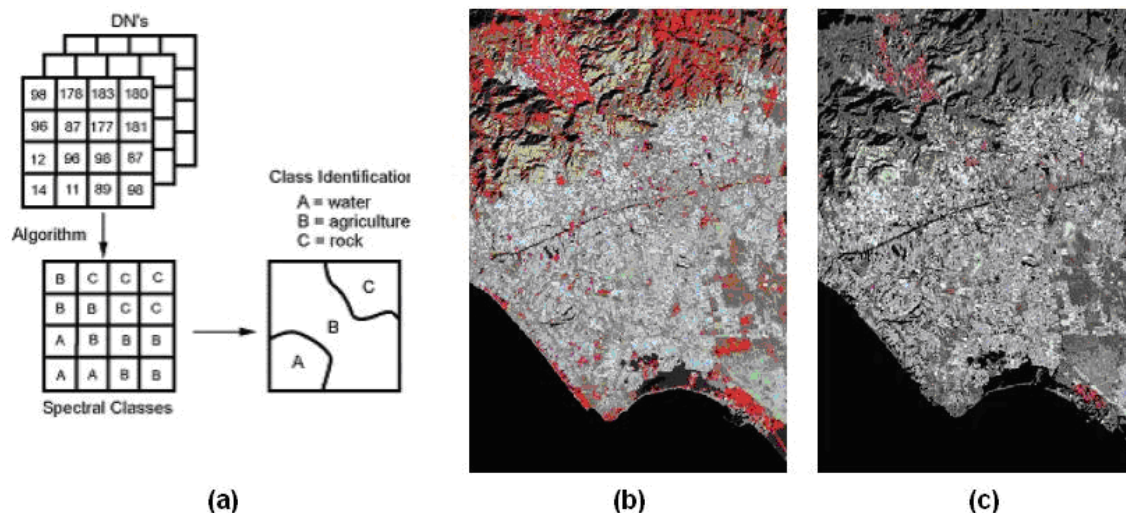


Figura 2.10: (a) Descripción del proceso general de clasificación. (b) Resultados de una clasificación supervisada. (c) Resultados de una clasificación no supervisada.

### 2.3. ALGORITMOS DE CLASIFICACIÓN NO SUPERVISADOS

En este apartado se explica el funcionamiento de dos algoritmos de clasificación de imágenes de satélite no supervisados, los clasificadores k-medias y isodata, que han sido muy utilizados en el ámbito de la teledetección.

#### 2.3.1. Clasificador k-medias

El algoritmo k-medias es uno de los clasificadores no supervisados más sencillos que existen. Se trata de un método de análisis de grupos que tiene como objetivo realizar una partición de un conjunto de elementos de entrada  $E$  en un número de clusters (o grupos)  $k$  determinado de antemano, de tal modo que cada cluster incorpora los elementos con la media más cercana. A continuación se muestran los pasos en los que se divide el algoritmo k-medias:

1. Se determina el número de clusters  $k$ .
2. Se seleccionan  $k$  centros iniciales tomados al azar.
3. Se asigna cada uno de los píxeles al cluster cuyo centro se encuentre más cercano.
4. Se calcula el valor de los centros con los nuevos píxeles asignados como el promedio de los elementos del cluster.
5. Si las fronteras de los clusters cambian significativamente, volver al paso 3.

Como se puede apreciar, en cada iteración se va refinando el proceso de clustering, de tal modo que se van agrupando los píxeles con similares características en la misma clase. Como salida, el algoritmo k-medias ofrece un conjunto formado por los centros de las  $k$  clases, así como los elementos de entrada  $E$  etiquetados con la clase correspondiente. La Tabla 2.1 muestra el pseudocódigo del algoritmo k-medias.

---

**Algoritmo k-medias** ( $E, k$ )

---

**Entrada:**

$E = \{e_1, e_2, \dots, e_n\}$ : conjunto de píxeles a clasificar  
 $k$ : número de clases

**Salida:**

$C = \{c_1, c_2, \dots, c_k\}$ : conjunto de centros de las clases  
 $L = \{l(e) \mid e = 1, 2, \dots, n\}$ : píxeles clasificados

---

```

01  foreach  $c_i \in C$  do
02       $c_i \leftarrow e_j \in E$ 
03  end
04  foreach  $e_i \in E$  do
05       $l(e_i) \leftarrow \minDistance(e_i, c_j) \mid j \in \{1..k\}$ 
06  end
07   $changed \leftarrow false;$ 
08  repeat
09      foreach  $c_i \in C$  do
10           $UpdateCluster(c_i)$ 
11      end
12      foreach  $e_i \in E$  do
13           $minDist \leftarrow \minDistance(e_i, c_j) \mid j \in \{1..k\}$ 
14          if  $minDist \neq l(e_i)$  then
15               $l(e_i) \leftarrow minDist$ 
16               $changed \leftarrow true$ 
17          end
18      end
19  until  $changed = true$ 

```

---

Tabla 2.1: Algoritmo de clasificación k-medias.

**2.3.2. Clasificador isodata**

El algoritmo isodata es una mejora del algoritmo k-medias, ya que permite no sólo especificar un número máximo de iteraciones para realizar el proceso de clustering, sino además establecer un umbral de convergencia que determine el porcentaje de píxeles mínimo que se debe modificar entre dos iteraciones para continuar con el algoritmo. Gracias a estas mejoras se consigue mejorar el proceso de creación de dicho agrupamiento. A continuación se muestran los pasos en los que se divide el algoritmo isodata:

1. *Se introducen los siguientes parámetros:  $k$  (número máximo de clusters),  $T$  (umbral de convergencia) y  $maxIters$  (número máximo de iteraciones).*
2. *Se determinan de forma arbitraria los centros de los  $k$  clusters.*
3. *Se calcula la distancia espectral de cada uno de los píxeles de la imagen a los centros. El píxel se asigna a la clase cuyo centro esté más cercano.*
4. *Con los nuevos píxeles asignados a clusters, se recalculan los nuevos centros.*

5. Si alguno de los clusters es demasiado pequeño se puede eliminar, disminuyendo el número de clusters determinado en un principio por el usuario. Si los centros de dos clusters están muy cercanos, éstos se pueden unir, y si un cluster contiene muchos objetos y su varianza en alguna variable es extremadamente grande, este cluster se divide en dos, siempre que no se supere el número máximo de clusters introducidos por el usuario,  $k$ .
6. Si no se ha alcanzado el número máximo de iteraciones  $maxIters$ , o el porcentaje de píxeles que se modifica entre dos iteraciones está por debajo del umbral de convergencia  $T$ , se vuelve al paso 3.

El algoritmo isodata es uno de los clasificadores no supervisados más utilizados, por sus buenos resultados. La Tabla 2.2 muestra el pseudocódigo del algoritmo isodata.

<b>Algoritmo isodata</b> ( $E, k, T, maxIters$ )	
<b>Entrada:</b>	
	$E = \{e_1, e_2, \dots, e_n\}$ : conjunto de píxeles a clasificar
	$k$ : número de clases
	$T$ : umbral de convergencia
	$maxIters$ : máximo de iteraciones
<b>Salida:</b>	
	$C = \{c_1, c_2, \dots, c_k\}$ : conjunto de centros de las clases
	$L = \{l(e) \mid e = 1, 2, \dots, n\}$ : píxeles clasificados
01	<b>foreach</b> $c_i \in C$ <b>do</b>
02	$c_i \leftarrow e_j \in E$
03	<b>end</b>
04	<b>foreach</b> $e_i \in E$ <b>do</b>
05	$l(e_i) \leftarrow \minDistance(e_i, c_j) \mid j \in \{1..k\}$
06	<b>end</b>
07	$iter \leftarrow 0$ ;
08	<b>repeat</b>
09	$changed \leftarrow 0$ ;
10	<b>foreach</b> $c_i \in C$ <b>do</b>
11	$UpdateCluster(c_i)$
12	<b>end</b>
13	<b>foreach</b> $e_i \in E$ <b>do</b>
14	$minDist \leftarrow \minDistance(e_i, c_j) \mid j \in \{1..k\}$
15	<b>if</b> $minDist \neq l(e_i)$ <b>then</b>
16	$l(e_i) \leftarrow minDist$
17	$changed \leftarrow changed + 1$
18	<b>end</b>
19	<b>end</b>
20	$iter ++$ ;
21	<b>until</b> $changed < T$ and $iter \leq maxIters$

Tabla 2.2: Algoritmo de clasificación isodata.

## 2.4. ALGORITMOS DE CLASIFICACIÓN SUPERVISADOS

En este apartado se explica el funcionamiento de tres algoritmos de clasificación de imágenes de satélite supervisados: mínima distancia, paralelepípedos y máxima verosimilitud.

### 2.4.1. Clasificador mínima distancia

Asigna a cada píxel la clase con la que existe menos distancia espectral respecto a su centroide, teniendo en cuenta todas las bandas que intervienen en el análisis. Para calcular la distancia entre el píxel y las clases se utiliza la fórmula de la distancia euclídea:

$$d_{ij}^{(k)} = \sqrt{\sum_{n=1}^{N_{bandas}} (X_{ij}^n - \bar{X}_{ij}^{(k)})^2} \quad (2.4)$$

donde:

- $k = 1, 2, 3, \dots, N$  son las clases.
- $X_{ij}^n (n = 1, 2, 3, \dots, N_{bandas})$  = valor de cada píxel en la banda  $n$ .
- $\bar{X}_{ij}^n (n = 1, 2, 3, \dots, N_{bandas})$  = valor del “centroide” (media aritmética) de la clase  $k$  en la banda  $n$ .
- $i, j$  = fila y columna del píxel de la imagen.

Cuando se han calculado todas las distancias entre el píxel y los centroides de las clases, el algoritmo asigna al píxel la clase más cercana siguiendo la siguiente fórmula:

$$clase(i, j) = \{k \mid d_{ij}^{(k)} = \text{minimo}\} \quad (2.5)$$

La Figura 2.11 muestra el proceso general de clasificación mediante el algoritmo mínima distancia en el espacio de características de la imagen de satélite.

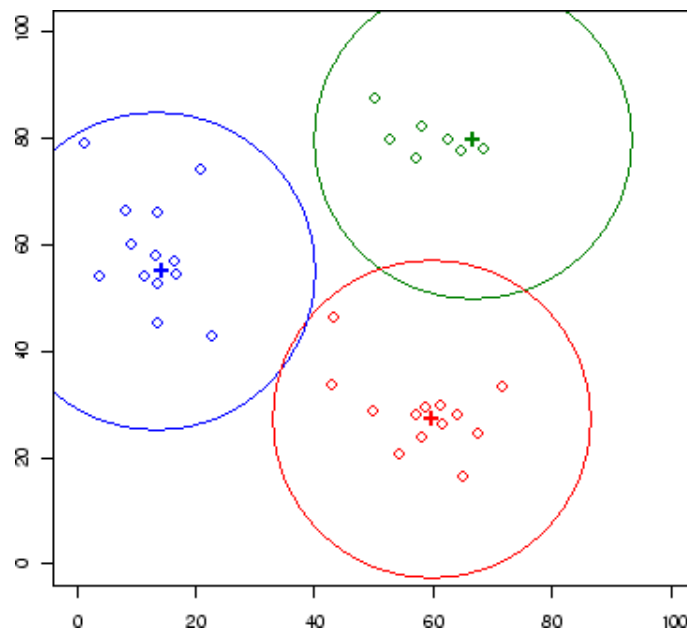


Figura 2.11: Clasificación mediante el algoritmo mínima distancia.

El clasificador de mínima distancia tiene como ventajas que todos los píxeles de la imagen son clasificados y que es rápido en ejecución. Sin embargo, tiene la inconveniencia de que es proclive a errores de comisión (asignar un píxel a una clase errónea) y que no considera la varianza de las clases.

### 2.4.2. Clasificador paralelepípedos

En este método, el usuario fija un área de dominio para cada categoría teniendo en cuenta sus valores de centralidad y dispersión. Posteriormente cada píxel es asignado a una clase si sus  $ND$  están dentro del área de dominio de esa clase, para todas las bandas que intervienen en la clasificación. El píxel  $x$  es asignado a la clase  $A$  si sus  $ND$  en las distintas bandas ( $ND_{x,k}$ ) están incluidos en el área de dominio de esa clase, tal y como se muestra en la siguiente fórmula:

$$\bar{ND}_{A,n} - RA,n \leq ND_{x,n} \leq \bar{ND}_{A,n} + RA,n \quad (2.6)$$

donde:

- $RA,n$ : rango de dispersión señalado para la categoría  $A$  en cada banda  $n$ .
- $n = 1, 2, 3, \dots, N$  son las bandas.

El clasificador de paralelepípedos tiene como ventajas que considera la dispersión de las clases y que es rápido en ejecución. Sin embargo, tiene la inconveniencia de que pueden quedar píxeles sin clasificar o algunos píxeles pueden clasificarse erróneamente en varias clases. Por lo tanto el algoritmo de paralelepípedos mejora los resultados obtenidos por el clasificador mínima distancia, ya que dispone de más información de cada una de las clases (no sólo la centralidad, sino también la dispersión), que aplica con el objetivo de mejorar el proceso final de clasificación.

La Figura 2.12 muestra el proceso general de clasificación mediante el algoritmo paralelepípedos en el espacio de características de la imagen de satélite.

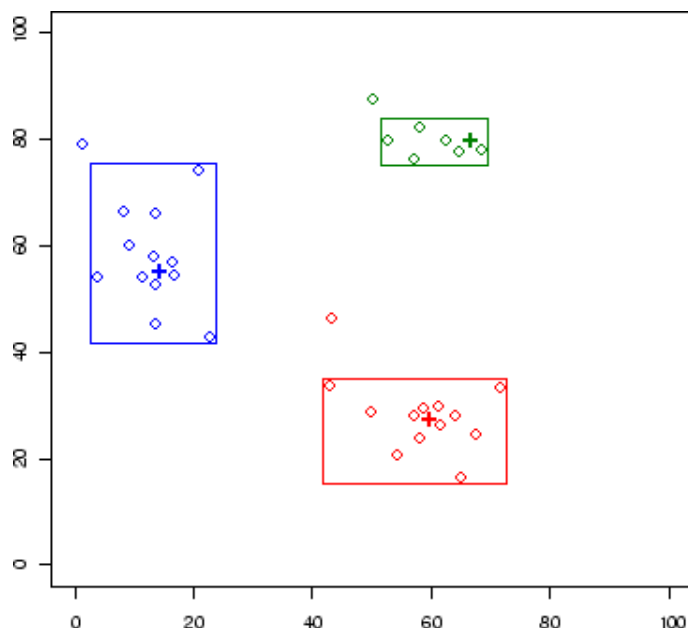


Figura 2.12: Clasificación mediante el algoritmo paralelepípedos.



### 2.4.3. Clasificador máxima verosimilitud

Este método considera que los  $ND$  de cada categoría se ajustan a una distribución normal. Esto nos permite describir esa categoría mediante una función de probabilidad, a partir de su vector de medias y matriz de covarianza. Así podemos calcular la probabilidad de que un determinado píxel pertenezca a una categoría: el cálculo se realiza para todas las categorías definidas y el píxel se clasifica en la clase a la cual tenga más probabilidad de pertenecer. Para calcular la probabilidad asociada a un cierto  $ND$  se necesita la media y la desviación típica de cada categoría. En el caso de trabajar con una sola banda la probabilidad se calcula de la siguiente forma:

$$p(x/A) = \frac{1}{\sqrt{2\pi\sigma_A^2}} e^{\{-(ND_x - \bar{ND}_A)^2/2\sigma_A^2\}} \quad (2.7)$$

donde:

- $p(x/A)$ : probabilidad de que un cierto píxel  $x$  (definido por  $ND_x$ ) pertenezca a la clase  $A$ .
- $\bar{ND}_A$ : media de la clase  $A$ .
- $\sigma_A^2$ : varianza de la clase  $A$ .

Una vez calculada la probabilidad para cada clase, un píxel es asignado a la clase  $A$  si:

$$p(x/A) \geq p(x/B) \quad (2.8)$$

$A \neq B$ , con  $B = 1, 2, 3, \dots, m$  y  $m$ =número de clases.

Este clasificador es el más fiable, ya que considera la variabilidad. Sin embargo, requiere un gran coste computacional y sobreclasifica las firmas. La Figura 2.13 muestra el proceso general de clasificación mediante el algoritmo máxima verosimilitud en el espacio de características de la imagen de satélite.

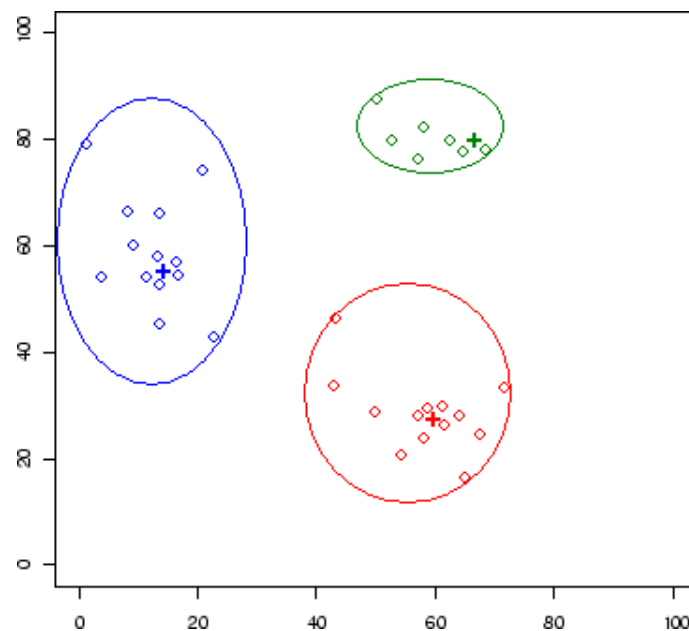


Figura 2.13: Clasificación mediante el algoritmo máxima verosimilitud.

## 2.5. RESUMEN Y CONCLUSIONES DEL CAPÍTULO

A pesar del elevado número de algoritmos supervisados y no supervisados que existen aplicados a la clasificación de imágenes de satélite, no existe en la actualidad uno que sea 100 % fiable, ya que todos tienen sus inconvenientes.

Existen algunos píxeles, a los que llamaré en este documento píxeles inciertos, que se encuentran a una distancia espectral casi equivalente de dos o más clases diferentes. Debido a este hecho, dichos píxeles pueden ser erróneamente clasificados si sólo tenemos en cuenta la información espectral de sus bandas. También podemos encontrarnos con píxeles de ruido en la imagen, aunque en un porcentaje de frecuencia mucho menor que los píxeles inciertos. Estos dos tipos de píxeles, con los algoritmos descritos en este capítulo, van a ser erróneamente clasificados. Por lo tanto es necesario utilizar algunas técnicas suplementarias para ayudar a dicho proceso de clasificación, como algoritmos contextuales.

Hasta el momento, las imágenes que incorporan una tasa de ruido por errores ocurridos durante el proceso de captación son tratadas mediante un determinado tipo de filtros antes de proceso de clasificación (pre-clasificación), y los píxeles inciertos son correctamente clasificados mediante algoritmos contextuales que se aplican a la clasificación obtenida previamente (post-clasificación). Sería interesante fusionar los tres procesos en uno solo: pre-clasificación, clasificación y post-clasificación. El presente trabajo documenta el trabajo de investigación que he realizado relacionado con este tema: clasificación de imágenes de satélite mediante autómatas celulares, con el objetivo de mejorar los algoritmos supervisados existentes y agrupar en un solo proceso la tarea de pre-clasificación, clasificación y post-clasificación de imágenes de satélite.

---

## CAPÍTULO 3

# AUTÓMATAS CELULARES

---



# Capítulo 3

## AUTÓMATAS CELULARES

### Contenidos

---

<b>3.1.</b>	INTRODUCCIÓN A LOS AUTÓMATAS CELULARES . . . . .	<b>43</b>
3.1.1.	Definición conceptual de autómatas celular . . . . .	43
3.1.2.	Definición formal de autómatas celular . . . . .	44
3.1.3.	Clasificación de los autómatas celulares . . . . .	45
<b>3.2.</b>	APLICACIONES DE LOS AUTÓMATAS CELULARES . . . . .	<b>46</b>
3.2.1.	Autómatas celulares unidimensionales. Notación de Stephen Wolfram . . . . .	46
3.2.2.	Autómatas celulares bidimensionales. Juego de la vida de Conway . . . . .	50
3.2.3.	Autómatas celulares aplicados a la teledetección . . . . .	51
<b>3.3.</b>	CARACTERIZACIÓN DE TEXTURAS MEDIANTE AUTÓMATAS CELULARES . . . . .	<b>52</b>
3.3.1.	Números decimales de las reglas de Stephen Wolfram . . . . .	52
3.3.2.	Caracterización de texturas de imágenes-AC . . . . .	53
3.3.3.	Caracterización de texturas naturales . . . . .	55
<b>3.4.</b>	RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .	<b>59</b>

---



**U**n autómata celular es un modelo matemático que evoluciona en pasos discretos y que se utiliza normalmente para simular sistemas de entidades naturales o artificiales que interactúan unas con otras evolucionando en el tiempo. Desde el origen de la teoría de autómatas celulares gracias a von Neumann, los autómatas celulares han sido muy usados en problemas relacionados con sistemas complejos, y en todo tipo de procesos de simulación. Este capítulo ofrece una introducción al modelo matemático del autómata celular, y está dividido en los siguientes apartados principales:

- Introducción a los autómatas celulares: en este apartado se ofrece una definición conceptual de autómata celular así como una posterior definición formal, y por último se muestra una clasificación básica de los autómatas celulares unidimensionales.
- Aplicaciones de los autómatas celulares: en este apartado se muestran las principales aplicaciones de los autómatas celulares unidimensionales así como la notación de Stephen Wolfram, las aplicaciones de los autómatas celulares bidimensionales así como el juego de la vida de Conway, y las aplicaciones más importantes de los autómatas celulares en el ámbito de la teledetección
- Caracterización de texturas mediante autómatas celulares: en este apartado se explica en primer lugar cómo obtener números decimales a partir de las reglas de Stephen Wolfram a través de un proceso de ingeniería inversa, a continuación cómo usar dichos números decimales obtenidos para caracterizar imágenes-AC y finalmente cómo caracterizar texturas naturales ampliando el procedimiento anterior.
- Resumen y conclusiones del capítulo: en este apartado se destacan los aspectos relacionados con los autómatas celulares en los que se ha centrado mi trabajo de investigación.

### 3.1. INTRODUCCIÓN A LOS AUTÓMATAS CELULARES

En este apartado se ofrece una definición conceptual de autómata celular así como una posterior definición formal, y por último se muestra una clasificación básica de los autómatas celulares unidimensionales.

#### 3.1.1. Definición conceptual de autómata celular

Un autómata celular es un modelo matemático que consiste en una malla de celdas distribuidas normalmente en forma de matriz que tiene las siguientes características básicas:

- *Estados*: cada celda toma un valor entero que corresponde con su estado actual. Hay un conjunto finito de estados posibles.
- *Vecindad*: conjunto de celdas que interactúan con la actual. La Figura 3.1 muestra los tipos de vecindad más comunes en autómatas celulares: 4 vecinos (vecindad de von Neumann), 8 vecinos (vecindad de Moore) y 24 vecinos (vecindad de Moore extendida).
- $f$ : función de transición que toma como entrada el estado de la celda actual y de la vecindad, y devuelve un nuevo estado para la celda actual.

- *Reglas*: la función de transición  $f$  utiliza un conjunto de reglas que especifican como cambian los estados de las celdas dependiendo de los valores de entrada que se tomen.
- *Iteraciones*: la función de transición  $f$  se aplica a cada celda de la malla a través de varias iteraciones.

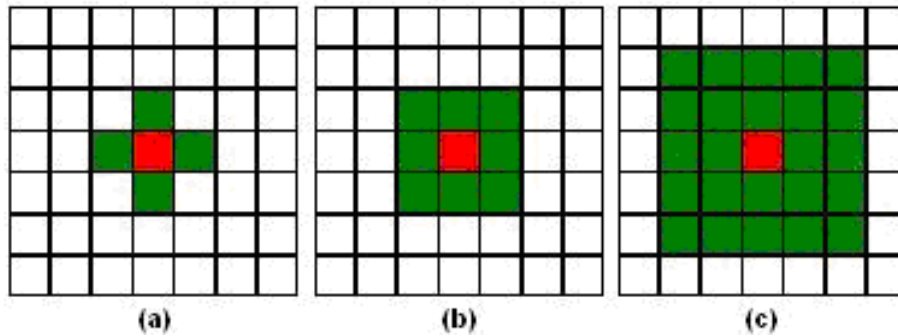


Figura 3.1: (a) Vecindad de von Neumann. (b) Vecindad de Moore. (c) Vecindad de Moore extendida.

Las celdas del autómata celular cambian su estado de manera discreta en el tiempo. En cada iteración, se comprueban todas las celdas y se aplican las reglas del autómata celular a través de la función de transición  $f$  a cada celda teniendo en cuenta no sólo su estado sino también el de su vecindad. Por lo tanto, los autómatas celulares sufren un proceso de evolución a lo largo de la ejecución de las distintas iteraciones, ya que los estados de las celdas cambian durante dicha ejecución. Desde este punto de vista, en los años recientes los autómatas celulares se han convertido en una herramienta muy potente para implementar y resolver problemas relacionados con procesos de simulación.

Los autómatas celulares se han aplicado para resolver una gran variedad de problemas de diversos tipos: creación de un modelo basado en autómatas celulares para crear terapias contra la infección del virus VIH [50], un sistema para comprender el crecimiento urbano [12], un modelo de migración de bancos de peces [48], autómatas celulares aplicados al mundo de los videojuegos [10], un modelo que simula una población infectada por una plaga periódica [19], un sistema basado en autómatas celulares para simular fluidos de lava [51], un modelo de ecosistema de vida artificial [61], un sistema para realizar simulaciones de enfermedades [13], un modelo que simula el flujo de agua en un canal de riego [38], un modelo para la competición y evolución de especies [14], sistemas para el estudio de la enfermedad del cáncer basados en autómatas celulares [53] [24], simulación del tráfico de vehículos en ciudades [1] o trenes en vías de ferrocarril [54], simulación de avalanchas de nieve [3], un sistema de iluminación modular [7] y nuevos sistemas criptográficos basados en autómatas celulares [22] [18].

### 3.1.2. Definición formal de autómata celular

**Definición 3.1 (Autómata celular)** *Un autómata celular es una séxtupla  $(d, r, Q, \#, V, f)$ , donde:*

- $d \mid d > 0$ : *dimensión espacial del autómata celular. La posición de cada celda se expresa mediante un vector de  $Z^d$ . Con  $d = 1$  tenemos un autómata unidimensional y con posición de las celdas en  $Z$ ; con  $d = 2$  tenemos un autómata bidimensional y con posición de las celdas en  $Z \times Z$ , etc.*



- $r$ : índice que marca el tamaño de la vecindad, es decir, el número de vecinos que interactúan con la celda actual.
- $Q$ : conjunto de estados que puede tomar cada celda. El conjunto de estados es finito e igual para todas las celdas.
- $\#$ : estado de  $Q$  llamado “quiescente”, que indica la ausencia de actividad
- $V$ : vector de vecindad que contiene  $r$  elementos distintos de  $Z^d$ .  $V \subset (Z^d)^r$
- $f$ : función de transición del autómata celular. Toma como argumentos de entrada los estados de la celda actual y de los vecinos, y devuelve un nuevo estado para la celda actual. La función de transición  $f$  utiliza un conjunto de reglas que especifican los procesos de cambio de los estados de las celdas. La función de transición  $f$  se aplica a cada celda de la matriz a través de varias iteraciones.

$$f : Q^{r+1} \rightarrow Q$$

$$q_i(t) = f(q_{i-r}(t-1), q_{i-r+1}(t-1), \dots, q_{i+r}(t-1)) \quad (3.1)$$

siendo  $q_i(t)$  el estado de la celda  $i$  en el tiempo  $t$ .

**Definición 3.2 (Configuración actual de un autómata celular)** Definimos la configuración actual de un autómata celular  $C$  a cualquier función:

$$\sigma : Z^d \rightarrow Q \quad (3.2)$$

**Definición 3.3 (Configuración siguiente de un autómata celular)** Definimos la configuración siguiente de un autómata celular en el tiempo  $t$  al resultado de aplicar las reglas del autómata a sus celdas con los estados de la configuración anterior  $t-1$ :

$$\sigma_t : f(\sigma_{t-1}) \quad (3.3)$$

Existen algunos trabajos que se han encargado de estudiar el aspecto más teórico de los autómatas celulares, donde se ofrecen desde un punto de vista matemático diversos teoremas y corolarios que se cumplen [30].

### 3.1.3. Clasificación de los autómatas celulares

Stephen Wolfram, basándose en el comportamiento dinámico de los autómatas celulares unidimensionales, los clasificó en cuatro tipos distintos [58] [59]:

- *Tipo 1*: el comportamiento del autómata celular es predecible, ya que todas las configuraciones evolucionan a un estado estable independientemente de cual sea la configuración inicial, con lo que los patrones desaparecen.
- *Tipo 2*: el autómata celular tiene un comportamiento periódico, ya que evoluciona hacia estructuras periódicas. Si se produce un cambio en un valor de la situación inicial sólo afecta a una región finita a su alrededor.
- *Tipo 3*: el comportamiento del autómata celular es caótico, y pequeños cambios en la configuración inicial provoca evoluciones distintas.
- *Tipo 4*: es el tipo más impreciso, donde se engloban sobre todo autómatas celulares que caen en un atractor después de pasar por una amplia fase evolutiva.

### 3.2. APLICACIONES DE LOS AUTÓMATAS CELULARES

En este apartado se muestran las principales aplicaciones de los autómatas celulares unidimensionales así como la notación de Stephen Wolfram, las aplicaciones de los autómatas celulares bidimensionales así como el juego de la vida de Conway, y las aplicaciones más importantes de los autómatas celulares en el ámbito de la teledetección.

#### 3.2.1. Autómatas celulares unidimensionales. Notación de Stephen Wolfram

Los autómatas celulares unidimensionales son aquellos que tienen una sola dimensión espacial, aunque realmente se representan de forma matricial. Cada generación del autómata celular unidimensional está representada por un renglón de la matriz, donde la primera generación corresponde con el primer renglón y las siguientes se ubican de manera correlativa [57]. A lo largo del presente documento, se denominará *imagen-AC* a la representación matricial de las distintas generaciones producidas por el autómata celular. A continuación se muestra un ejemplo de autómata celular unidimensional:

$$C = (1, 2, \{0, 1\}, 0, V, f), \text{ donde:}$$

$$V = \{-1, 1\}$$

Se trata de un autómata celular con una dimensión espacial unidimensional y 2 vecinos por celda. Cada celda puede tomar 2 estados posibles: 0 y 1. El estado “quiescente” es el 0, y la vecindad de cada celda corresponde a la celda de la izquierda y de la derecha. La función de transición  $f$  del autómata celular se muestra a continuación:

$$f(q_{i-1}(t-1), q_i(t-1), q_{i+1}(t-1)) = (q_{i-1}(t-1) + q_{i+1}(t-1)) \bmod 2 \quad (3.4)$$

La Tabla 3.1 muestra la evolución que experimenta dicho autómata celular unidimensional a través de 4 iteraciones, partiendo del estado inicial  $t=0$ .

t=0	0	0	0	0	1	0	0	0	0
t=1	0	0	0	1	0	1	0	0	0
t=2	0	0	1	0	0	0	1	0	0
t=3	0	1	0	0	0	0	0	1	0
t=4	1	0	0	0	0	0	0	0	1

Tabla 3.1: Evolución del autómata celular unidimensional del ejemplo.

El precursor de los autómatas celulares unidimensionales fue el científico Stephen Wolfram [60], creador del software Mathematica. Creó una numeración específica para este tipo de autómata celular. Para ello, en primer lugar se representa cada función de transición como una terna o vector formado por el estado actual, y los estados de las dos celdas vecinas.

$$\delta_i: (q_h, [e_1, e_2]) \Rightarrow q_t \text{ es equivalente a } \delta_i: ([e_1, q_h, e_2]) \Rightarrow q_t$$

Donde  $q_h$  es el estado de la celda actual, y  $e_1, e_2$  son los estados de las celdas adyacentes. A continuación se muestra un ejemplo concreto de función de transición. Si tomamos todas las combinaciones posibles, podemos representar dicha función de transición de la siguiente manera:

$$\begin{aligned} \delta_1: (0, [0, 0]) &\Rightarrow 0 \text{ es equivalente a } \delta_1:(0, 0, 0) \Rightarrow 0 \\ \delta_2: (0, [0, 1]) &\Rightarrow 1 \text{ es equivalente a } \delta_2:(0, 0, 1) \Rightarrow 1 \\ \delta_3: (1, [0, 0]) &\Rightarrow 0 \text{ es equivalente a } \delta_3:(0, 1, 0) \Rightarrow 0 \\ \delta_4: (1, [0, 1]) &\Rightarrow 1 \text{ es equivalente a } \delta_4:(0, 1, 1) \Rightarrow 1 \\ \delta_5: (0, [1, 0]) &\Rightarrow 1 \text{ es equivalente a } \delta_5:(1, 0, 0) \Rightarrow 1 \\ \delta_6: (0, [1, 1]) &\Rightarrow 1 \text{ es equivalente a } \delta_6:(1, 0, 1) \Rightarrow 1 \\ \delta_7: (1, [1, 0]) &\Rightarrow 1 \text{ es equivalente a } \delta_7:(1, 1, 0) \Rightarrow 1 \\ \delta_8: (1, [1, 1]) &\Rightarrow 0 \text{ es equivalente a } \delta_8:(1, 1, 1) \Rightarrow 0 \end{aligned}$$

El vector generado por cada combinación anterior se puede expresar como un número binario de 3 dígitos, y podemos asignar a cada valor binario su estado final. A continuación podemos crear un vector de estados finales ordenados y expresarlo en sistema de numeración decimal, de tal forma que el ejemplo anterior quedaría expresado de la siguiente manera:

$$R = [0,1,0,1,1,1,1,0] \text{ (notación binaria)} \quad R = 94 \text{ (notación decimal)}$$

Por lo tanto, según la notación de Stephen Wolfram, la función de transición del ejemplo anterior se puede resumir en la regla R94 obtenida al pasar a decimal el número binario formado por los estados finales ordenados producidos por las 8 combinaciones anteriormente realizadas. Como partimos de un número formado por 8 dígitos binarios, podremos crear un total de 256 reglas básicas. Cuando aplicamos una regla específica a un renglón de estados iniciales, va surgiendo un nuevo renglón en cada iteración, que unidos forman la matriz de generaciones a la que hemos llamado imagen-AC, es decir, se forma una imagen aplicando un autómata celular unidimensional a un estado inicial. La Figura 3.2 muestra la imagen-AC obtenida mediante la regla R94 tras aplicar 15 iteraciones.

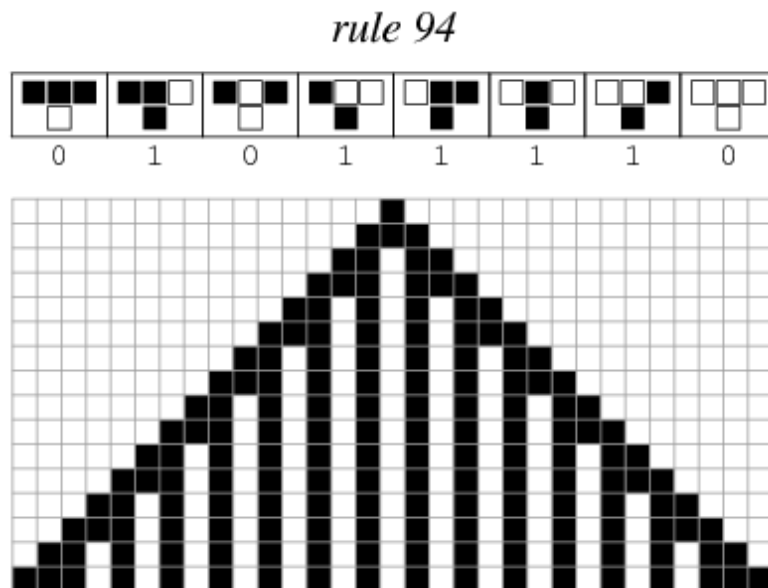


Figura 3.2: Imagen-AC generada por la regla R94.

Estas reglas, además, pueden expresarse mediante fórmulas matemáticas. Si comenzamos con una sola celda negra, podemos codificar en binario las generaciones sucesivas y comprobamos que forman el siguiente patrón: 1, 111, 11011, 1110111, 110101011, etc., que en sistema de numeración decimal corresponde al patrón: 1, 7, 27, 119, 427, ... Dicha sucesión puede expresarse a través de la siguiente fórmula matemática:

$$a(n) = \begin{cases} 1 & \text{si } n = 0 \\ 7 & \text{si } n = 1 \\ \frac{1}{6}(10 + 11 \cdot 4^n) & \text{si } n > 1 \text{ es impar} \\ \frac{1}{3}(1 + 5 \cdot 4^n) & \text{si } n > 0 \text{ es par} \end{cases} \quad (3.5)$$

La Figura 3.3 muestra la imagen-AC creada por otra regla de Stephen Wolfram, en este caso la regla R54.

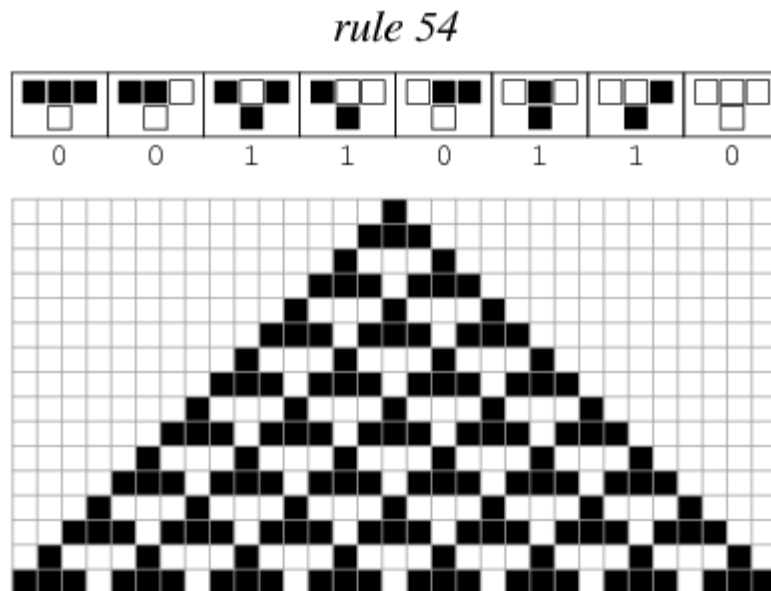


Figura 3.3: Imagen-AC generada por la regla R54.

También podemos comprobar en la siguiente fórmula que dicha regla sigue un patrón numérico decimal en la generación de nuevos estados:

$$a(n) = \begin{cases} \frac{7}{15}(4^{n+1} - 1) & \text{si } n \text{ es impar} \\ \frac{1}{15}(4^{n+2} - 1) & \text{si } n \text{ es par} \end{cases} \quad (3.6)$$

En la Figura 3.4 se pueden observar las reglas más conocidas con la notación expuesta de Stephen Wolfram, así como las imágenes-AC resultantes de cada regla tras aplicar 15 iteraciones. Se puede observar los patrones que crea cada regla concreta. La Figura 3.5 muestra las imágenes-AC más conocidas generadas por las reglas de Stephen Wolfram.

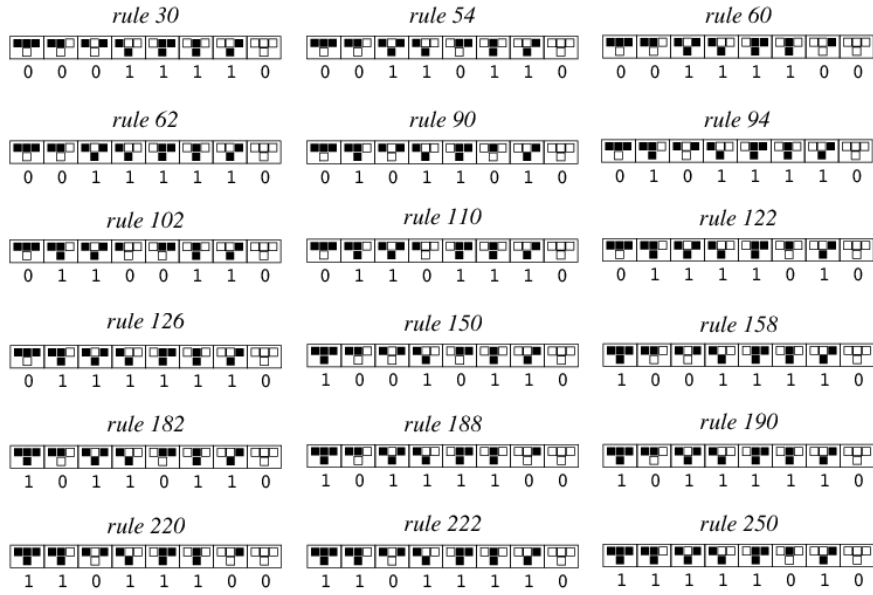


Figura 3.4: Reglas que generan las imágenes-AC más conocidas.

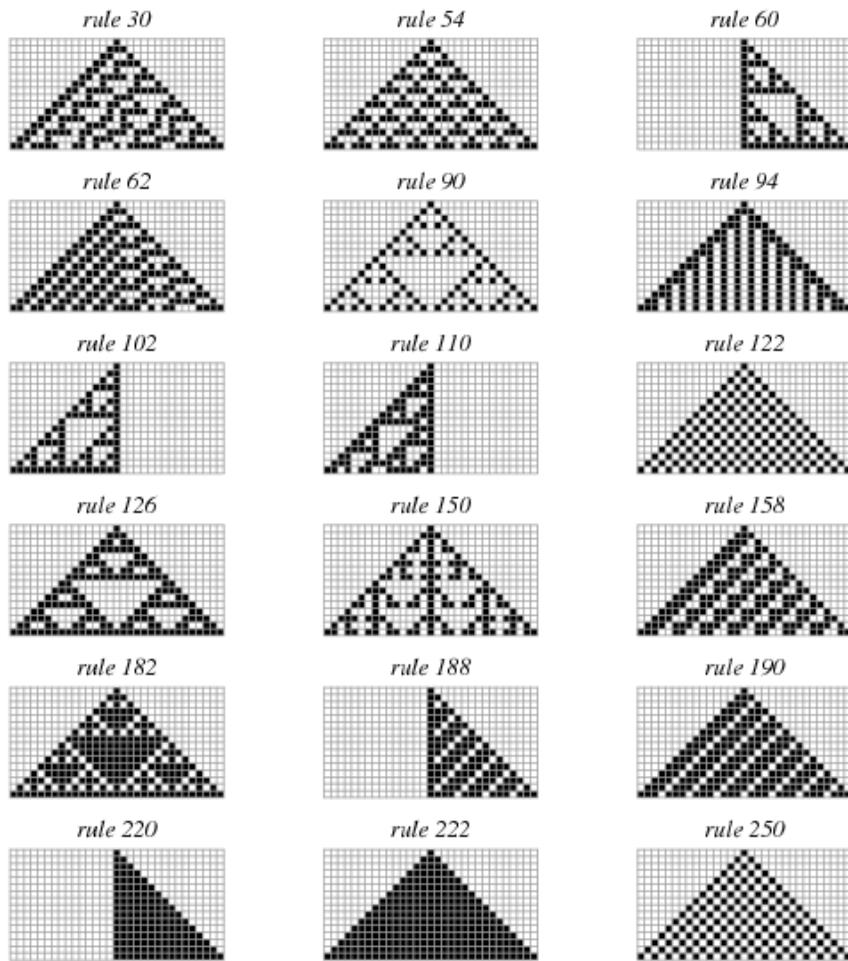


Figura 3.5: Imágenes-AC más conocidas de Stephen Wolfram.

### 3.2.2. Autómatas celulares bidimensionales. Juego de la vida de Conway

Los autómatas celulares bidimensionales son aquellos en los que las celdas se sitúan en un universo plano de dos dimensiones, es decir, una cuadrícula, de tal modo que cada celda tiene 8 vecinos adyacentes.

Una de las aplicaciones más conocidas de autómatas celulares bidimensionales es el llamado “juego de la vida” diseñado por el matemático John Horton Conway en 1970, y publicado en la columna de juegos matemáticos de la revista *Scientific American* por Martin Gardner [23]. Dicha publicación fue de gran interés para diversos campos de la ciencia, sobre todo en el ámbito de las matemáticas y economía, debido a la gran variabilidad que ofrece el algoritmo en la evolución de los patrones.

Como su propio nombre indica, el juego de la vida es un algoritmo basado en un autómata celular bidimensional que simula el proceso mismo de la vida, en el que cada celda corresponde a un ser vivo que, como tal, tiene la posibilidad de nacer, reproducirse, mantenerse con vida y morir por soledad o superpoblación. La definición del autómata celular que implementa el juego de la vida de Conway es la siguiente:

$$C = (2, 8, \{0, 1\}, 0, V, f), \text{ donde:}$$

$$V = \{(-1, -1), (0, -1), (1, -1), (-1, 0), (1, 0), (-1, 1), (0, 1), (1, 1)\}$$

Se trata de un autómata celular con una dimensión espacial bidimensional y 8 vecinos por celda. Cada celda puede tomar 2 estados posibles: 0 y 1, que corresponde con los estados *muerta* y *viva* respectivamente. El estado “quiescente” es el 0, y la vecindad de cada celda es de tipo Moore. La función de transición  $f$  del autómata celular se muestra a continuación:

$$f(q_0(t-1), q_1(t-1), \dots, q_8(t-1)) = \begin{cases} q_0 & \text{si } \sum_{i=1}^8 q_i(t-1) = 2 \\ 1 & \text{si } \sum_{i=1}^8 q_i(t-1) = 3 \\ 0 & \end{cases} \quad (3.7)$$

Como se puede observar en la primera opción de la función de transición del autómata celular, cuando una celda posee 2 vecinos con vida, ésta permanece en el mismo estado en el que se encontraba (proceso de estabilidad). La segunda opción de la función de transición indica que, cuando una celda está rodeada por 3 celdas vivas, nace (proceso de reproducción). La tercera opción indica que, en cualquier otro caso, la celda muere (procesos de soledad y superpoblación). La Figura 3.6 muestra un ejemplo de cada tipo de proceso.

La Figura 3.7 muestra una simulación del juego de la vida de Conway a través de 75 iteraciones del autómata celular.

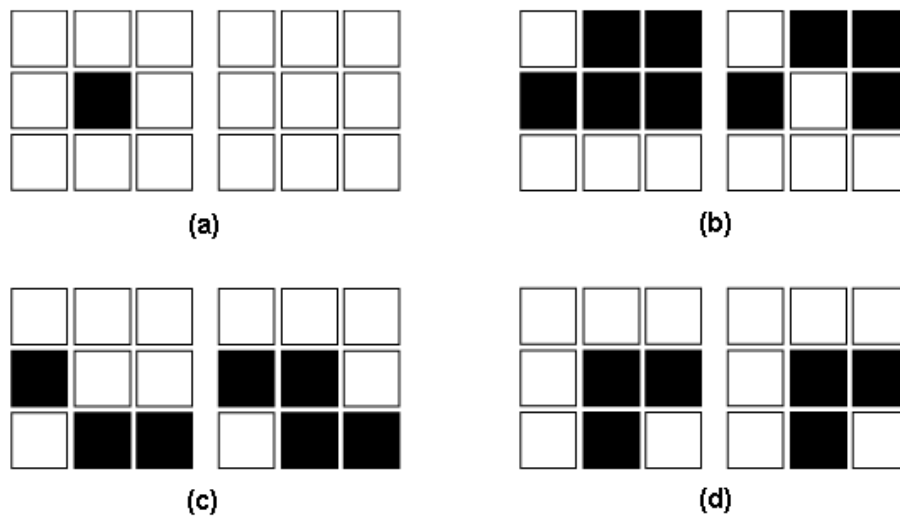


Figura 3.6: (a) Soledad: una celda viva con menos de 2 celdas vivas adyacentes muere. (b) Sobrepoblación: una celda viva con más de 3 celdas vivas adyacentes muere. (c) Reproducción: una celda muerta con 3 celdas vivas adyacentes nace. (d) Estabilidad: una celda viva con dos celdas vivas adyacentes permanece viva.

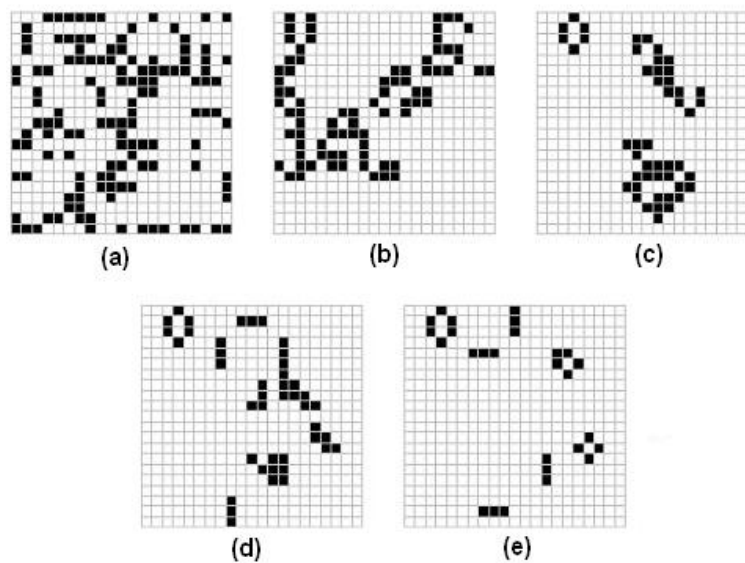


Figura 3.7: (a) Condiciones iniciales. (b) Iteración 10. (c) Iteración 25. (d) Iteración 50. (e) Iteración 75.

### 3.2.3. Autómatas celulares aplicados a la teledetección

Cuando aplicamos los autómatas celulares a las imágenes de satélite, consideramos que cada píxel de la imagen es una celda de un autómata celular bidimensional con una distribución espacial en forma de matriz, y la vecindad de cada píxel o celda del autómata suele estar formada por los 8 píxeles vecinos (vecindad de Moore), aunque dicha vecindad puede variar dependiendo del objetivo que se pretende conseguir y puede estar formado por sólo 4 píxeles vecinos (vecindad de von Neumann) o incluso los 24 píxeles de alrededor (vecindad de Moore extendida).

Los cambios en los estados de las celdas del autómata celular (píxeles de la imagen) ocurren de forma discreta en el tiempo. En cada iteración del autómata celular se comprueba, para cada celda, su estado y los estados de su vecindad, para posteriormente aplicar las reglas de la función de transición  $f$  y obtener de esta forma el nuevo estado de cada celda del autómata. Por lo tanto los autómatas celulares tienen un proceso de evolución, ya que todas las celdas están cambiando continuamente de estado a través de la función de transición a lo largo de las distintas iteraciones.

Desde este punto de vista, los autómatas celulares se han convertido en los últimos años en una herramienta de gran potencia en el ámbito de la teledetección, y han sido utilizados sobre todo en aplicaciones que implican desarrollar algún tipo de simulación en la imagen de satélite, como pueden ser simulaciones medioambientales, resolución de fenómenos sociales complejos, aplicaciones avanzadas de visión artificial, criptografía de información digital o inteligencia artificial en juegos matemáticos.

En el ámbito medioambiental se han realizado numerosos trabajos fusionando la teledetección con la técnica de autómatas celulares: simulaciones avanzadas del proceso de derretimiento de la nieve con influencia de varios factores externos [33], modelización dinámica de sistemas de vegetación [6], estudio de la propagación de brotes de cólera mediante la mediciones indirectas del terreno [36], simulaciones de propagación de incendios forestales [43] [44] [29], modelado de procesos de deforestación con elevada biocomplejidad [42] o simulaciones dinámicas del empleo de la tierra [32] [40].

También se han realizado trabajos aplicando los autómatas celulares a la teledetección para modelar distintos fenómenos sociales de elevada complejidad [56], como por ejemplo el control del ruido del tráfico de una ciudad [46], o incluso técnicas avanzadas para la generación de mallas en el diseño urbano y arquitectónico [11].

Dentro del ámbito de la visión artificial, los autómatas celulares se han aplicado para la obtención de mejoras en imágenes, implementando filtros de reducción de ruido, y en la detección de bordes [45].

### 3.3. CARACTERIZACIÓN DE TEXTURAS MEDIANTE AUTÓMATAS CELULARES

En este apartado se explica en primer lugar cómo obtener números decimales a partir de las reglas de Stephen Wolfram a través de un proceso de ingeniería inversa, a continuación se expone cómo usar dichos números decimales obtenidos para caracterizar imágenes-AC y finalmente se explica cómo caracterizar texturas naturales ampliando el procedimiento anterior.

#### 3.3.1. Números decimales de las reglas de Stephen Wolfram

La interpretación computerizada de imágenes digitales es una de las ramas más importantes de la informática y tiene como objetivo la adquisición de información útil a partir de imágenes procedentes de diversos campos de la ciencia como la medicina, las aplicaciones industriales, el medio ambiente y la teledetección a través de técnicas de reconocimiento de patrones e inteligencia artificial. La percepción visual de la rugosidad o suavidad de la superficie de los objetos, que es una característica conocida como textura, se puede utilizar como información adicional para la caracterización de dichas imágenes digitales. A pesar de la importancia de esta característica, en la actualidad y en nuestro conocimiento, no existe una definición satisfactoria y precisa para la textura [27].



El proceso de detección y análisis de la textura en imágenes puede dividirse en dos grandes categorías: estructurales y estadísticos [26]. Los métodos de análisis estructurales asumen que las texturas se componen de primitivas de textura [28] [55]. Los métodos estadísticos se basan en valores estadísticos de los niveles de gris de la imagen, y han sido ampliamente estudiados desde la aparición del artículo de R. Haralick [27], que propuso varias características basadas en la matriz de coocurrencia. Otros métodos más recientes han investigado la combinación de técnicas estadísticas y estructurales [55], enfoques basados en modelos relacionados con la transformada wavelet [34], los campos aleatorios de Markov (MRF) y técnicas fractales [35].

Desde el proyecto I+D SOLERES se ha implementado un nuevo método para caracterizar diferentes tipos de texturas basado en las propiedades de los autómatas celulares lineales. El método propuesto se deriva de las propiedades de las imágenes-AC producidas por un modelo de autómata celular lineal de 3 celdas, que es uno de los modelos más sencillos en este campo. El objetivo principal de este estudio es mostrar la viabilidad de las técnicas basadas en autómatas celulares para caracterizar texturas naturales. Las pruebas se han realizado con éxito en imágenes binarias proporcionadas por el Álbum Brodatz.

En el marco del proyecto SOLERES, y con el objetivo de crear un nuevo método para caracterizar la textura de las imágenes digitales basado en autómatas celulares, se ha desarrollado un algoritmo de caracterización de texturas basado en las propiedades de las imágenes-AC generadas por las reglas de 3 celdas descritas en apartados anteriores [31]. Como se expuso en apartados anteriores, dadas dos líneas consecutivas de una imagen-AC con valores 0 y 1, la segunda línea toma sus valores de la aplicación de una regla de 3 celdas específica a la primera línea. Generalizando dicho proceso, cada celda con valor 1 de una línea cualquiera se obtiene aplicando la regla de 3 celdas a las tres celdas vecinas de la línea anterior, que a su vez representan un número binario que puede tomar los valores 0-7 al cambiar a base decimal. La Tabla 3.2 muestra el proceso de adquisición de los números decimales de la regla de 3 celdas en el que se obtienen los números decimales 1, 2, 5 y 6. A partir de dichos valores decimales deducimos que la segunda fila se ha obtenido a partir de la primera mediante la aplicación de una regla lineal de 3 celdas que en la notación binaria tiene un 1 en las posiciones 1, 2, 5 y 6, es decir, la regla es  $01100110_2 = 102_{10}$ , conocida como la regla R102 de Stephen Wolfram. Como se puede observar, el procedimiento no proporciona información acerca de los sitios ubicados en las fronteras de la imagen o cuando hay un 0 presente, sólo con el valor 1. Cabe señalar que, en cierto sentido, el método está relacionado con las propiedades de inversión de los autómatas celulares [52]. De este modo, realizando un proceso similar al de ingeniería inversa en la rama de la programación informática, se puede obtener la regla que describe una imagen-AC a partir de dos líneas consecutivas de la imagen.

Primera línea	1	0	1	0	1	1	1	0	1	1	0	1	0	0	1
Segunda línea	1	1	1	1	0	0	1	1	0	1	1	1	0	1	0
Número de 3-celdas	-	5	2	5	-	-	6	5	-	6	5	2	-	1	-

Tabla 3.2: Adquisición de los números decimales de la regla de 3 celdas.

### 3.3.2. Caracterización de texturas de imágenes-AC

Cuando este tipo de análisis se extiende a una imagen-AC completa, tenemos tantas líneas como el número de filas tiene la imagen, y para calcular la regla de generación correspondiente es necesario calcular la cantidad de números decimales de la imagen completa.

Si aplicamos el método propuesto a la imagen-AC que se muestra en la Figura 3.8 (a), la cantidad resultante de números decimales, o histograma correspondiente, es el que se muestra en la Figura 3.8 (b), donde se puede observar que los únicos números decimales de la gráfica son 0, 5, 6 y 7, que en la notación binaria es  $11100001_2 = 225_{10}$ . La Figura 3.8 muestra la imagen y el histograma de las primitivas pertenecientes a la imagen-AC resultado de la regla R225.

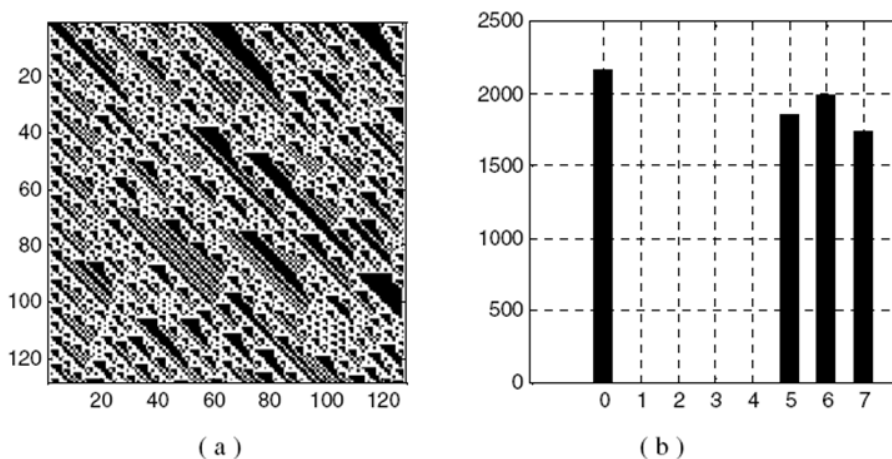


Figura 3.8: (a) Imagen-AC obtenida con la regla R225. (b) Histograma de números decimales obtenidos aplicando el método propuesto.

El histograma de los números decimales obtenido de la aplicación del procedimiento propuesto juega un papel importante en la caracterización de la textura de una imagen binaria, ya sea una imagen de textura natural o una imagen-AC. Ya que el histograma mencionado está asociado al número de primitivas de una imagen-AC, lo vamos a definir como histograma de primitivas (PH). La Figura 3.9 muestra las primitivas de textura correspondientes a la imagen generada por la regla R102.

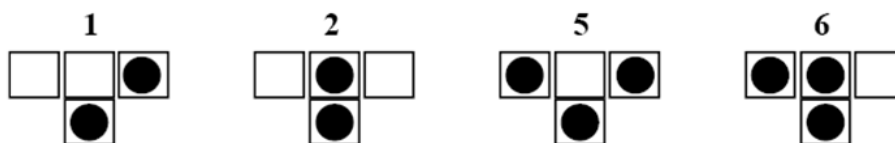


Figura 3.9: Primitivas de textura correspondientes a la imagen-AC generada por la regla R102.

El método propuesto también se puede aplicar a una imagen compuesta por varias imágenes-AC, como se muestra en la Figura 3.10(izquierda). Esta imagen, de tamaño 256x256 píxeles (o celdas), es en realidad la composición de cuatro imágenes-AC de 128x128 píxeles cada una, generadas por las reglas: R12 y R123 en la fila superior, y las reglas R60 y R90 en la fila inferior. El análisis de la textura de la imagen compuesta se ha realizado aplicando el método propuesto por bloques de 16x16 píxeles, con un total de 256 bloques en la imagen. Cada bloque de salida tiene un nivel de gris asociado a la primitiva de textura presente en el PH. La Figura 3.10(derecha) muestra el resultado de construir una imagen con los bloques resultantes en sus respectivas posiciones.

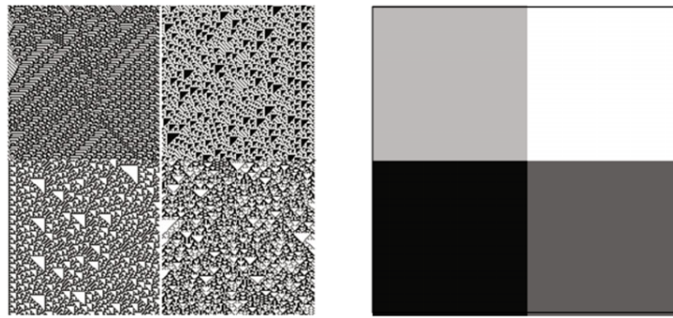


Figura 3.10: (Izquierda) Imagen (256x256) compuesta por 4 sectores de distintas texturas: sectores superiores (R121 y R153), sectores inferiores (R60 y R90). (Derecha) Imagen en escala de grises resultante de la aplicación del método propuesto.

### 3.3.3. Caracterización de texturas naturales

Para aplicar este método al análisis de texturas naturales, que es nuestro principal interés, vamos a suponer que los patrones de textura mostrado por estas imágenes son el resultado de un proceso similar al que se produce en la imágenes-AC. Sin embargo, con el fin de llevar a cabo el método propuesto a las texturas naturales, tenemos que superar previamente dos problemas:

- En primer lugar, el número de niveles de gris en este tipo de imágenes es normalmente mayor que dos. Este problema puede solucionarse aplicando una binarización a la imagen en estudio, con la admisión de una pérdida natural de detalles en la textura de la imagen.
- En segundo lugar, cuando calculamos el PH de la imagen binaria correspondiente aparece un variado número de primitivas, pero con una frecuencia de ocurrencia dispersa. De hecho, en el PH que resulta de estas imágenes no pueden estar presentes las 256 primitivas, por lo tanto, sólo se deben elegir las más importantes de todas las existentes.

La Figura 3.11 muestra 3 imágenes del álbum Brodatz y sus imágenes binarias correspondientes.

Hasta ahora, cuando hemos hablado sobre el cálculo del PH, nos hemos referido al PH que resulta de aplicar el método propuesto fila por fila a una textura determinada. A este histograma le vamos a llamar PH horizontal (HPH). Con el objetivo de tener en cuenta otras primitivas, que no están presentes en el HPH, vamos a realizar el mismo proceso en la dirección vertical de la imagen (es decir, columna por columna). A este histograma le llamamos PH vertical (VPH). Por supuesto, la información adicional proporcionada por el VPH mejorará la caracterización de la textura en estudio, pero eso sí, a costa de aumentar el número de opciones para la selección de las mejores primitivas. El método que se ha propuesto desde el proyecto SOLERES para caracterizar la textura de las imágenes naturales se pueden resumir en el algoritmo en pseudocódigo que se muestra en la Tabla 3.3.

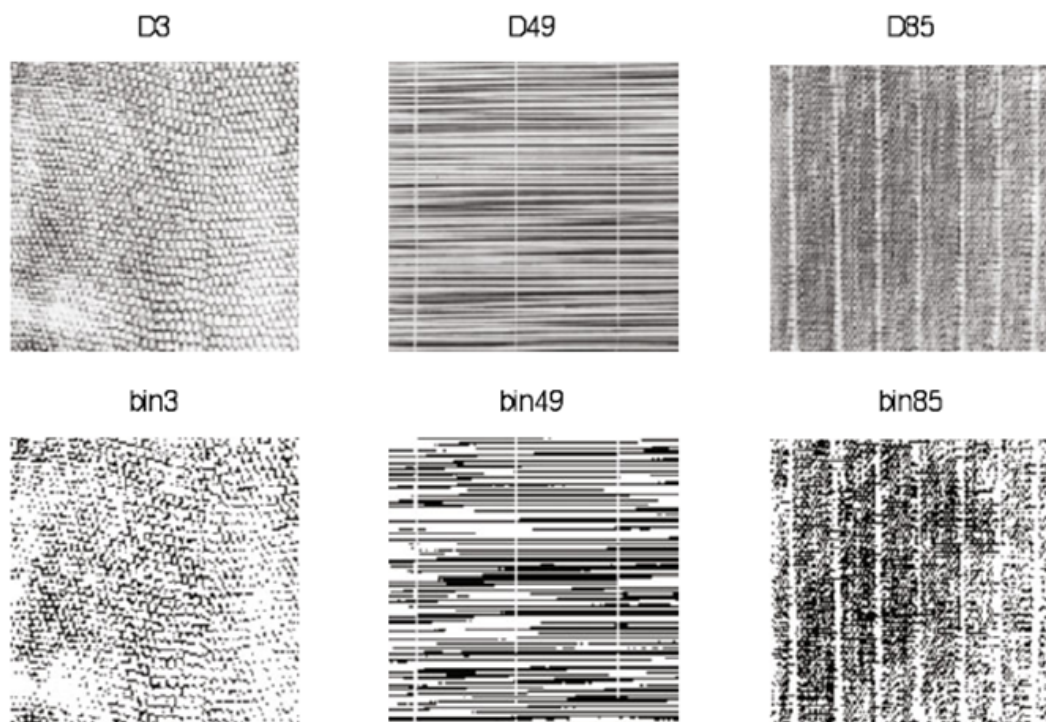


Figura 3.11: Arriba: 3 imágenes del álbum Brodatz: D3, D49 y D85. Abajo: imágenes binarias correspondientes.

---

### Algoritmo de caracterización de texturas naturales con autómatas celulares

---

#### Entrada:

*imagenOriginal* = imagen con una textura natural

#### Salida:

*imagen - AC* = imagen generada por reglas de Stephen Wolfram

---

- 01 Partimos de una imagen con una textura natural.
  - 02 Se aplica un proceso de binarización a la imagen con la textura natural. Suponemos que la imagen resultante (representada por unos y ceros) ha sido generada a partir de un autómata celular lineal.
  - 03 Se calcula el HPH de la imagen binaria a través del procedimiento propuesto
  - 04 Se seleccionan las primitivas más importantes del HPH (de 1 a 8).
  - 05 Se genera la imagen-AC correspondiente a la regla que surge del paso anterior (características horizontales).
  - 06 Se calcula el VPH de la imagen binaria a través del procedimiento propuesto
  - 07 Se seleccionan las primitivas más importantes del VPH (de 1 a 8).
  - 08 Se genera la imagen-AC correspondiente a la regla que surge del paso anterior (características verticales).
  - 09 Se genera una imagen que caracteriza la textura natural de la que partimos en el paso 1. Esta imagen es una composición (normalmente la suma) de las imágenes-AC obtenidas en los pasos 5 y 8.
- 

Tabla 3.3: Algoritmo de caracterización de texturas naturales mediante autómatas celulares.

Como ejemplo ilustrativo se ha aplicado el algoritmo propuesto a la versión binaria de la imagen Brodatz D85, denotada en la Figura 3.11 como bin85. En la Figura 3.12(a) se muestra el HPH obtenido, y en la Figura 3.12(b) el VPH.

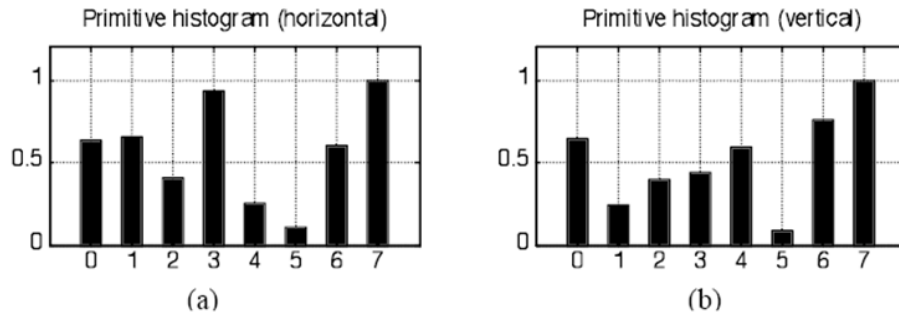


Figura 3.12: Histogramas de primitivas normalizados para la imagen Brodatz binaria D85. (a) HPH (dirección horizontal). (b) VPH (dirección vertical).

Los pasos 4 y 7 del algoritmo propuesto para caracterizar texturas naturales mediante autómatas celulares son los más difíciles de aplicar, ya que en la etapa actual de la investigación, no se ha encontrado un método práctico para superar este problema. El número de primitivas horizontales y verticales es elegido convenientemente mediante la observación de cada histograma. Una vez elegidas las primitivas más importantes, se obtiene el número de la regla para cada sentido y es posible general los “indicadores” horizontal y vertical de la textura original. Estos indicadores de textura se muestran en la Figura 3.13.

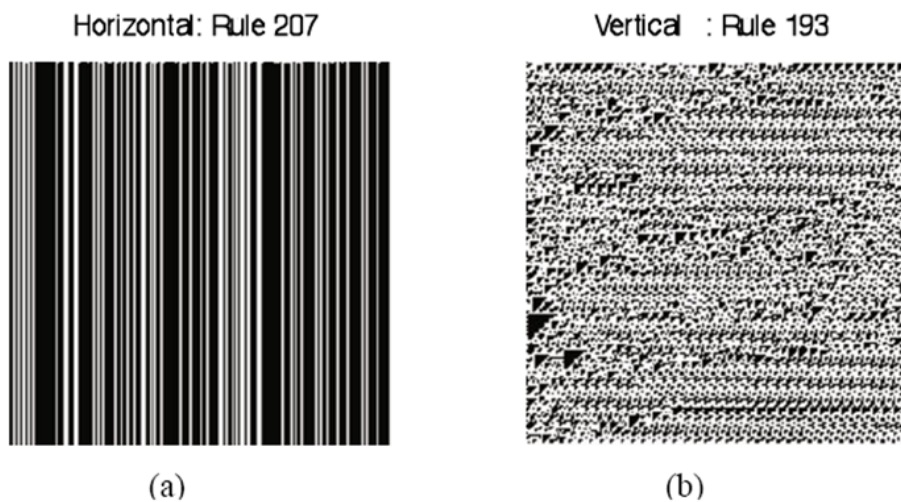


Figura 3.13: (a) Imágenes-AC obtenidas como resultado de elegir las primitivas más importantes de los histogramas anteriores. Del histograma vertical se han tomado 6 primitivas y del horizontal 3. Aplicando el método propuesto se han obtenido las reglas 207 (horizontal) y 193 (vertical).

Las Figuras 3.14, 3.15 y 3.16 muestran los resultados obtenidos en la aplicación del algoritmo propuesto a las imágenes de Brodatz D3, D49 y D85. La composición que se utilizan en estos ejemplos es una simple suma de cada componente.



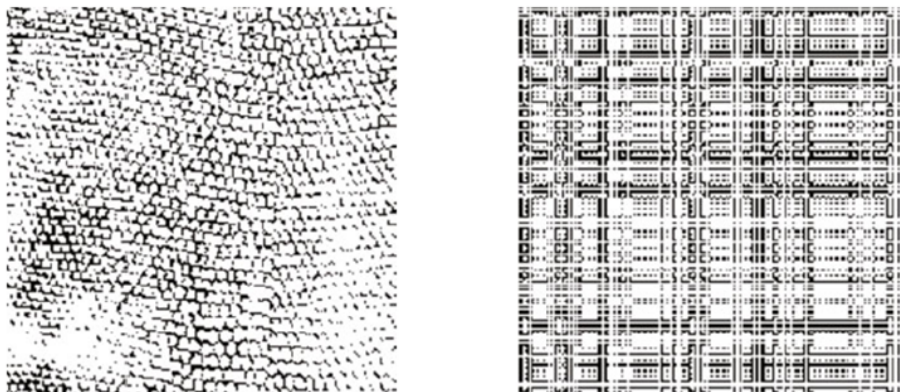


Figura 3.14: Izquierda: imagen bin3 (256x256). Derecha: aproximación de la imagen bin3 obtenida por la composición de las imágenes-AC R95 (H) y R95 (V) obtenidas usando 6 componentes de cada histograma primitivo (H y V).

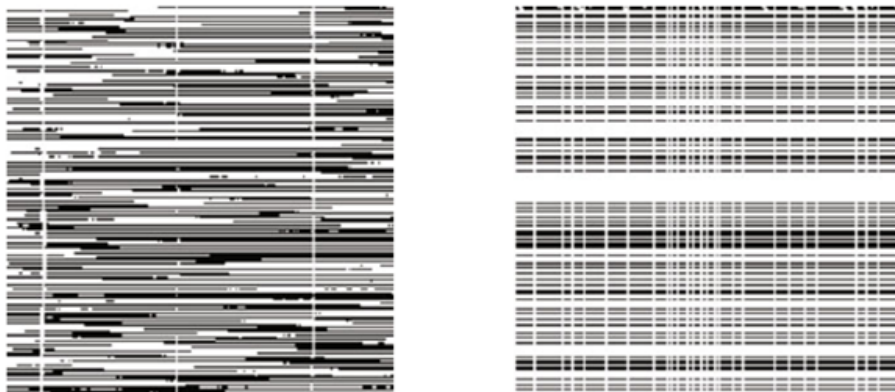


Figura 3.15: Izquierda: imagen bin49 (256x256). Derecha: aproximación de la imagen bin49 obtenida por la composición de las imágenes-AC R217 (H) y R76 (V) obtenidas usando 5 componentes del HPH y 3 componentes del VPH.

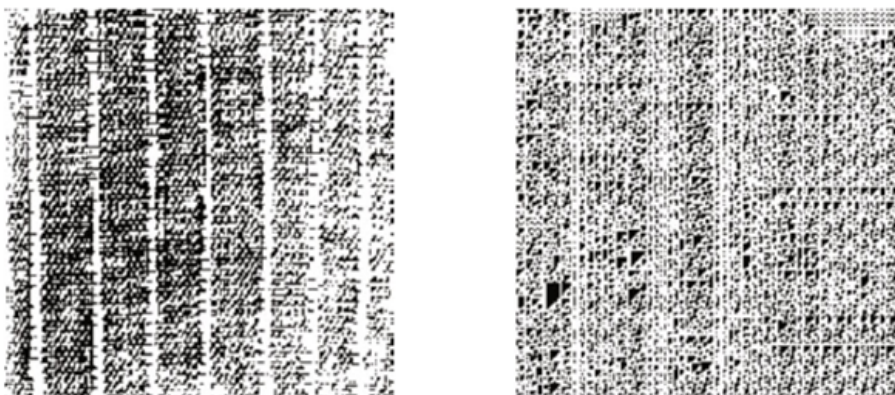


Figura 3.16: Izquierda: imagen bin85 (256x256). Derecha: aproximación de la imagen bin85 obtenida por la composición de las imágenes-AC R207 (H) y R193 (V) obtenidas usando 6 componentes del HPH y 3 componentes del VPH.

### 3.4. RESUMEN Y CONCLUSIONES DEL CAPÍTULO

Los autómatas celulares han resultado ser una herramienta muy útil para resolver todo tipo de problemas relacionados con procesos de simulación. En el ámbito concreto de la teledetección, los autómatas celulares se han utilizado sobre todo para simular fenómenos naturales. Desde el proyecto I+D SOLERES se ha creado un algoritmo de caracterización de texturas naturales mediante autómatas celulares, como se ha comentado en este capítulo. Además los primeros resultados han demostrado que los autómatas celulares pueden ser de gran utilidad a la hora de caracterizar texturas naturales, por lo que se ha ampliado el ámbito de aplicación de este modelo matemático.

No obstante, como se comenta al final del capítulo 2, aunque en el ámbito de la teledetección existen numerosos algoritmos supervisados y no supervisados de clasificación de imágenes de satélite, ninguno ofrece una fiabilidad del 100% en los resultados obtenidos. Desde el proyecto I+D SOLERES se ha propuesto un nuevo algoritmo supervisado de clasificación de imágenes de satélite basado en autómatas celulares, que mejora los resultados de algunos algoritmos de clasificación clásicos. En el siguiente capítulo se ofrece una descripción detallada de este nuevo algoritmo que se ha obtenido como fruto de mi período de investigación, y alrededor del cual gira todo el trabajo del presente documento.





---

## CAPÍTULO 4

# CLASIFICACIÓN CON AUTÓMATAS CELULARES

---



# Capítulo 4

## CLASIFICACIÓN CON AUTÓMATAS CELULARES

### Contenidos

---

<b>4.1.</b>	<b>PROBLEMAS DE LOS ALGORITMOS DE CLASIFICACIÓN SUPERVISADOS . . .</b>	<b>65</b>
4.1.1.	Inconvenientes de los algoritmos de clasificación supervisados . . .	65
4.1.2.	Cálculo de la calidad del proceso de clasificación . . . . .	66
4.1.3.	Análisis de los problemas en la clasificación supervisada clásica . .	68
<b>4.2.</b>	<b>CLASIFICACIÓN DE IMÁGENES DE SATÉLITE CON AUTÓMATAS CELULARES</b>	<b>74</b>
4.2.1.	Trabajo previo existente . . . . .	74
4.2.2.	Objetivos principales del algoritmo ACA . . . . .	74
4.2.3.	Algoritmo ACA v1.0: estados y reglas . . . . .	76
<b>4.3.</b>	<b>DESCRIPCIÓN DEL ALGORITMO ACA . . . . .</b>	<b>78</b>
4.3.1.	Arquitectura del algoritmo ACA . . . . .	78
4.3.2.	Pseudocódigo del algoritmo ACA . . . . .	79
4.3.3.	Simulación del algoritmo ACA . . . . .	82
<b>4.4.</b>	<b>RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .</b>	<b>83</b>

---



**E**n la actualidad existen diversos tipos de algoritmos de clasificación supervisada de imágenes de satélite, como por ejemplo los clasificadores mínima distancia, paralelepípedos y máxima verosimilitud. Sin embargo, dichos algoritmos ofrecen una serie de inconvenientes, de hecho no existe en la actualidad un clasificador 100% eficiente. Desde el marco del proyecto I+D SOLERES: *Un Sistema de Información Espacio-Temporal para la Gestión Medioambiental basado en Redes Neuronales, Agentes y Componentes Software*, se ha desarrollado un nuevo algoritmo de clasificación de imágenes de satélite basado en autómatas celulares que es descrito detalladamente a lo largo del presente documento, al que se le ha dado el nombre de ACA (Algorithm based on Cellular Automata) [20] [21]. Este nuevo tipo de clasificador de imágenes de satélite mejora en varios aspectos los resultados obtenidos por otros algoritmos clásicos, como se describe en el siguiente capítulo. Este capítulo ofrece una descripción detallada del algoritmo ACA, y está dividido en los siguientes apartados principales:

- Problemas de los algoritmos de clasificación supervisados: en este apartado se ofrece una descripción de los inconvenientes de los algoritmos de clasificación supervisados clásicos, se realiza el cálculo de la calidad del proceso de clasificación de un ejemplo práctico y se analiza el origen de los problemas de calidad en la clasificación supervisada clásica.
- Clasificación de imágenes de satélite con autómatas celulares: en este apartado se cita el trabajo previo existente relacionado con la clasificación de imágenes con autómatas celulares, se enumeran los objetivos que pretende alcanzar el algoritmo ACA y se muestran los estados y reglas de la versión ACA 1.0.
- Descripción del algoritmo ACA: en este apartado se explica en profundidad la arquitectura completa del algoritmo ACA, se expone el pseudocódigo del algoritmo y se muestra una simulación del mismo.
- Resumen y conclusiones del capítulo: en este apartado se destacan los aspectos más importantes del algoritmo ACA para clasificación de imágenes de satélite mediante autómatas celulares.

## 4.1. PROBLEMAS DE LOS ALGORITMOS DE CLASIFICACIÓN SUPERVISADOS

En este apartado se ofrece una descripción de los inconvenientes de los algoritmos de clasificación supervisados clásicos, se realiza el cálculo de la calidad del proceso de clasificación de un ejemplo práctico y se analiza el origen de los problemas de calidad en la clasificación supervisada clásica.

### 4.1.1. Inconvenientes de los algoritmos de clasificación supervisados

En el capítulo 2 se detallan las características fundamentales de los algoritmos de clasificación supervisados más conocidos (mínima distancia, paralelepípedos y máxima verosimilitud). En cada sección correspondiente, además, se citan las ventajas e inconvenientes de cada algoritmo de clasificación en concreto.

En este apartado nos centraremos en estudiar las ventajas e inconvenientes de dichos algoritmos, para poder mejorar su funcionamiento posteriormente mediante el uso de autómatas celulares. La Tabla 4.1 muestra un resumen de las ventajas e inconvenientes que presentan dichos algoritmos de clasificación supervisados clásicos.

	Ventajas	Inconvenientes
Mínima distancia	Todos los píxeles clasificados Rápido de ejecutar	Proclive a errores de comisión No considera la varianza
Paralelepípedos	Considera la dispersión Rápido de ejecutar	Quedan píxeles sin clasificar Píxeles en varias clases
Máxima verosimilitud	Método muy fiable Considera la variabilidad	Elevado coste computacional Sobreclasifica las firmas

Tabla 4.1: Ventajas e inconvenientes de los algoritmos de clasificación supervisados más usados: mínima distancia, paralelepípedos y máxima verosimilitud.

Si nos centramos en los inconvenientes de dichos algoritmos, podemos observar que algunos de ellos se pueden solucionar con el uso apropiado de algoritmos contextuales que mejoran la clasificación obtenida. Es decir, algoritmos que no sólo utilizan la información espectral de cada píxel para asignarlo a una clase determinada, sino también los valores de los vecinos de alrededor para aquellos píxeles que se encuentran espectralmente cercanos a varias clases y por lo tanto son más complejos de clasificar. A este tipo de píxeles les vamos a llamar en este documento *píxeles inciertos* ya que, si un algoritmo de clasificación se basa sólo en sus valores espectrales, no sabrá a ciencia cierta a qué clase pertenece. Los píxeles inciertos son los causantes del inconveniente *proclive a errores de comisión* del algoritmo mínima distancia y del inconveniente *píxeles en varias clases* del algoritmo paralelepípedos, por ejemplo.

En la siguiente sección vamos a centrarnos en los inconvenientes del algoritmo mínima distancia a través de un ejemplo práctico de clasificación de una imagen de satélite mediante dicho algoritmo, para posteriormente analizar los errores que comete durante el proceso de clasificación.

#### 4.1.2. Cálculo de la calidad del proceso de clasificación

En este apartado vamos a calcular el índice de calidad obtenido en un ejemplo práctico de clasificación con el algoritmo mínima distancia. Las pruebas de clasificación se han llevado a cabo en una imagen Landsat de 7 bandas de una franja regional de las provincias Almería-Granada (sureste de España), de un tamaño de 300x300 píxeles y una resolución espacial de 30mx30m.

Los píxeles de dicha imagen han sido etiquetados en 8 clases distintas, obtenidas previamente mediante “trabajo de campo” realizado por parte del equipo de investigadores de ecología del proyecto I+D SOLERES, utilizando para ello mapas ecológicos y correlacionándolos posteriormente con imágenes de satélite [16] [17].

La Figura 4.1 muestra la zona de estudio clasificada mediante trabajo de campo y la Figura 4.2 muestra la clasificación supervisada obtenida mediante el algoritmo mínima distancia, basándose en un conjunto de muestras inicial tomado previamente de cada una de las 8 clases especificadas por el trabajo de campo realizado.

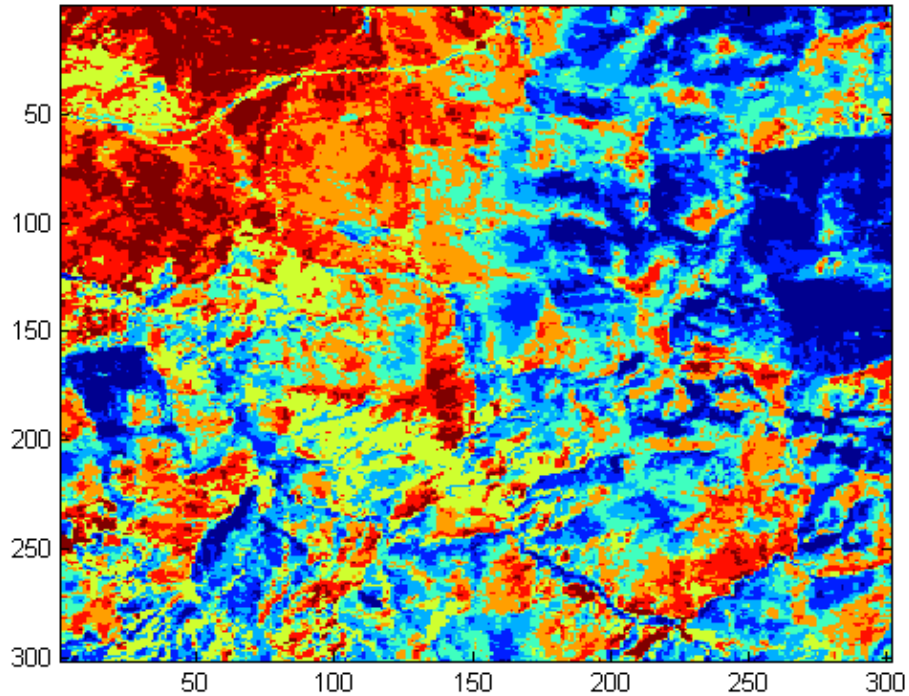


Figura 4.1: Clasificación obtenida mediante trabajo de campo.

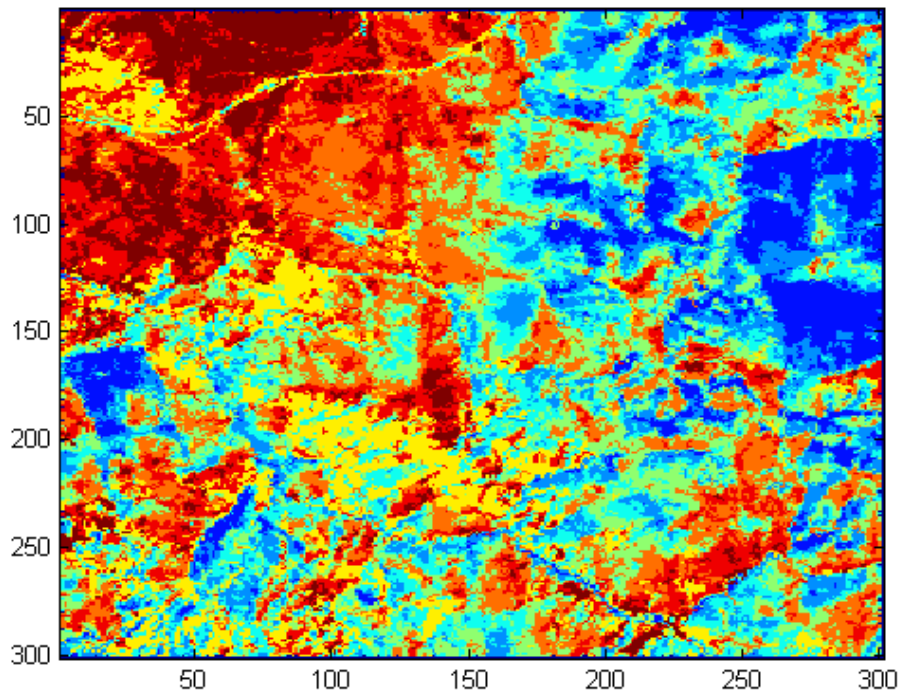


Figura 4.2: Clasificación obtenida mediante el algoritmo mínima distancia.

Aunque a primera vista puede parecer que los resultados obtenidos en ambos procesos de clasificación son similares, hemos utilizado los cálculos de una matriz de confusión para determinar qué píxeles han sido clasificados de manera correcta, y qué píxeles se han clasificado erróneamente. La Tabla 5.1 muestra la matriz de confusión obtenida.

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8
Clase 1	<b>0</b>	0	0	0	0	0	0	0
Clase 2	789	<b>6080</b>	355	0	0	0	0	0
Clase 3	1247	0	<b>9447</b>	532	0	0	0	0
Clase 4	1547	0	2	<b>11998</b>	242	0	0	0
Clase 5	1555	0	0	52	<b>12827</b>	3	27	0
Clase 6	1027	0	0	47	281	<b>8330</b>	35	1
Clase 7	1513	0	0	0	250	12	<b>13050</b>	0
Clase 8	1396	0	0	0	0	66	381	<b>11242</b>

Tabla 4.2: Matriz de confusión del algoritmo mínima distancia.

Los píxeles bien clasificados son aquellos que aparecen en la diagonal principal (destacados en negrita en la matriz de confusión). El resto de píxeles han sido clasificados de manera errónea. En total, un 80 % de los píxeles han sido bien clasificados, y un 20 % mal clasificados.

En la siguiente sección analizamos en detalle los errores que ha cometido el algoritmo supervisado mínima distancia en el proceso de clasificación, para posteriormente indicar las pautas necesarias para obtener mejores resultados mediante el uso de autómatas celulares.

### 4.1.3. Análisis de los problemas en la clasificación supervisada clásica

En este apartado vamos a analizar los errores cometidos por el algoritmo mínima distancia en el proceso de clasificación, con el objetivo de poder corregirlos posteriormente mediante el uso de autómatas celulares. Para ello hemos comenzado obteniendo una lista de 200 píxeles mal clasificados por el algoritmo mínima distancia, tomados al azar. La Tabla 4.3 muestra la lista de 200 píxeles mal clasificados tomados al azar, donde se muestra la posición  $[x, y]$  de cada píxel, y a continuación la clase a la que pertenece dicho píxel (según la clasificación mediante trabajo de campo) y la clase que se le ha asignado erróneamente (según el algoritmo mínima distancia).

Como se puede observar en dicha tabla, se han destacado en negrita los píxeles que poseen la relación de clases “5, 7” (píxeles que pertenecen a la clase 5 y se han clasificado erróneamente en la clase 7). De los 200 píxeles mal clasificados tomados al azar, 45 son de este tipo (un 22.5 % del total). Eso nos lleva a deducir que ambas clases se encuentran muy próximas entre sí desde el punto de vista espectral, y por lo tanto existe cierta ambigüedad en el proceso de clasificación.

Tomamos a modo de ejemplo dos píxeles del tipo “5, 7”, concretamente los píxeles 2 y 3 de la Tabla 4.3, cuya posición es  $[3, 161]$  y  $[3, 162]$ . Las Figuras 4.3 y 4.4 muestran la clasificación obtenida mediante trabajo de campo y mínima distancia respectivamente. Si nos fijamos con detenimiento en los dos píxeles que estamos estudiando, nos damos cuenta de que están rodeados en su mayoría por píxeles de la clase 5, en lugar de píxeles de la clase 7.



Píxel	Clases	Píxel	Clases	Píxel	Clases	Píxel	Clases
[2, 218]	<b>5, 7</b>	[93, 92]	7, 8	[174, 32]	4, 5	<b>[238, 259]</b>	<b>5, 7</b>
[3, 161]	<b>5, 7</b>	[94, 109]	7, 8	[174, 253]	7, 8	[239, 34]	7, 8
[3, 162]	<b>5, 7</b>	[94, 248]	4, 5	[175, 159]	3, 5	[242, 173]	6, 3
[3, 293]	4, 5	[94, 249]	4, 5	[175, 169]	4, 5	[243, 296]	7, 8
[6, 241]	1, 2	[95, 296]	1, 2	[178, 146]	8, 6	[246, 97]	7, 8
[7, 9]	5, 3	[96, 123]	7, 8	[179, 32]	4, 5	[247, 259]	7, 8
[7, 245]	1, 2	[97, 282]	4, 5	[182, 21]	2, 4	[248, 288]	1, 2
[8, 206]	2, 4	<b>[99, 146]</b>	<b>5, 7</b>	[182, 169]	4, 5	<b>[250, 82]</b>	<b>5, 7</b>
[8, 207]	2, 4	[103, 108]	4, 5	[185, 30]	7, 8	[251, 117]	4, 5
[8, 271]	2, 4	[104, 108]	4, 5	[186, 187]	4, 5	[251, 136]	2, 4
[10, 190]	2, 4	[104, 134]	7, 8	[186, 267]	4, 5	<b>[252, 157]</b>	<b>5, 7</b>
[11, 164]	4, 5	<b>[104, 276]</b>	<b>5, 7</b>	[189, 300]	4, 5	[253, 134]	2, 4
<b>[12, 168]</b>	<b>5, 7</b>	[115, 270]	2, 4	<b>[190, 199]</b>	<b>5, 7</b>	<b>[254, 116]</b>	<b>5, 7</b>
[14, 294]	4, 5	[116, 191]	4, 5	<b>[194, 50]</b>	<b>5, 7</b>	[255, 161]	7, 8
<b>[18, 144]</b>	<b>5, 7</b>	[117, 191]	4, 5	[195, 274]	2, 4	[256, 81]	8, 6
[20, 169]	4, 5	[118, 35]	8, 6	[195, 291]	2, 4	[257, 81]	8, 6
[21, 169]	4, 5	[118, 194]	2, 4	<b>[196, 232]</b>	<b>5, 7</b>	[258, 300]	2, 4
[26, 145]	7, 8	[118, 260]	2, 4	[196, 291]	2, 4	[260, 107]	7, 3
[27, 251]	4, 5	[119, 260]	2, 4	[197, 57]	5, 3	[260, 241]	7, 8
<b>[28, 285]</b>	<b>5, 7</b>	[123, 275]	3, 5	[197, 73]	4, 5	[260, 242]	7, 8
[29, 174]	4, 5	[126, 38]	4, 5	[197, 300]	1, 2	[262, 71]	2, 4
[34, 230]	2, 4	[126, 282]	1, 2	[198, 290]	4, 5	<b>[262, 205]</b>	<b>5, 7</b>
<b>[34, 250]</b>	<b>5, 7</b>	<b>[127, 239]</b>	<b>5, 7</b>	[198, 300]	1, 2	[263, 17]	4, 5
[36, 213]	2, 4	<b>[127, 240]</b>	<b>5, 7</b>	[199, 16]	2, 4	[263, 48]	2, 4
<b>[38, 206]</b>	<b>5, 7</b>	[133, 178]	4, 5	<b>[199, 83]</b>	<b>5, 7</b>	[264, 70]	2, 4
[43, 236]	4, 5	<b>[133, 225]</b>	<b>5, 7</b>	<b>[199, 198]</b>	<b>5, 7</b>	[270, 210]	7, 8
[43, 260]	4, 5	[137, 156]	4, 5	[200, 22]	4, 5	[271, 55]	4, 5
<b>[44, 286]</b>	<b>5, 7</b>	[141, 90]	8, 6	[200, 26]	4, 5	[274, 121]	2, 4
<b>[45, 278]</b>	<b>5, 7</b>	<b>[141, 174]</b>	<b>5, 7</b>	[203, 76]	2, 4	[278, 166]	8, 6
[50, 230]	2, 4	<b>[142, 174]</b>	<b>5, 7</b>	[203, 238]	4, 5	<b>[279, 279]</b>	<b>5, 7</b>
<b>[51, 243]</b>	<b>5, 7</b>	[144, 202]	4, 5	<b>[204, 61]</b>	<b>5, 7</b>	[280, 93]	4, 5
<b>[52, 197]</b>	<b>5, 7</b>	<b>[145, 231]</b>	<b>5, 7</b>	[205, 18]	2, 4	[281, 45]	2, 4
[55, 52]	7, 8	[148, 134]	7, 8	[205, 59]	4, 5	[282, 226]	8, 6
<b>[56, 251]</b>	<b>5, 7</b>	[149, 186]	5, 3	<b>[209, 208]</b>	<b>5, 7</b>	[283, 176]	7, 8
[59, 167]	2, 4	[150, 81]	4, 5	[209, 241]	4, 5	<b>[284, 10]</b>	<b>5, 7</b>
[59, 238]	2, 4	[150, 122]	4, 5	[214, 96]	3, 5	[284, 168]	4, 5
[59, 268]	5, 3	[150, 224]	2, 4	[215, 120]	4, 5	<b>[285, 242]</b>	<b>5, 7</b>
[60, 282]	4, 5	[150, 250]	2, 4	[215, 263]	7, 8	[286, 65]	4, 5
[61, 85]	7, 8	[150, 251]	2, 4	<b>[217, 19]</b>	<b>5, 7</b>	[287, 65]	4, 5
[62, 21]	7, 8	[151, 224]	2, 4	[217, 64]	8, 6	[290, 110]	4, 5
[63, 144]	7, 8	[151, 225]	2, 4	[218, 192]	4, 5	<b>[290, 180]</b>	<b>5, 7</b>
[69, 102]	7, 8	[152, 225]	2, 4	<b>[220, 88]</b>	<b>5, 7</b>	[294, 65]	2, 4
[69, 247]	4, 5	[153, 240]	4, 5	<b>[220, 89]</b>	<b>5, 7</b>	<b>[294, 223]</b>	<b>5, 7</b>
[70, 235]	4, 5	<b>[153, 244]</b>	<b>5, 7</b>	[224, 40]	7, 8	[294, 257]	7, 8
[71, 117]	7, 8	<b>[154, 120]</b>	<b>5, 7</b>	<b>[225, 26]</b>	<b>5, 7</b>	[297, 120]	4, 5
[76, 192]	4, 5	[154, 240]	4, 5	[229, 12]	6, 3	[298, 75]	2, 4
[76, 211]	7, 8	[157, 281]	1, 2	<b>[229, 28]</b>	<b>5, 7</b>	[298, 120]	4, 5
[78, 67]	7, 8	[162, 197]	4, 5	[230, 202]	4, 5	[299, 263]	4, 5
[79, 246]	4, 5	[163, 283]	1, 2	[231, 226]	4, 5	[300, 212]	1, 2
[93, 17]	7, 8	[169, 210]	4, 5	[233, 233]	7, 8	[300, 217]	7, 8

Tabla 4.3: Lista de 200 píxeles mal clasificados tomados al azar. En negrita se destacan los píxeles que poseen la relación de clases “5, 7”.

Row	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170
0	8	7	5	5	2	1	2	6	5	2	2	2	4	3	3	3	6	8	7
1	8	7	7	5	4	2	2	3	5	4	4	4	5	4	4	7	6	8	8
2	8	9	8	7	7	5	4	4	5	5	5	7	3	4	3	7	7	8	8
3	9	9	9	9	8	7	7	7	4	5	5	7	3	3	3	7	8	8	8
4	10	9	9	9	8	8	8	8	8	5	3	4	3	9	6	5	8	8	8
5	9	9	9	9	8	8	9	8	3	3	3	6	6	8	7	7	8	8	8
6	8	8	9	9	8	8	6	6	3	3	6	4	7	8	5	5	7	8	8
7	8	8	8	9	8	7	3	3	3	3	7	4	4	2	4	5	7	8	8
8	8	8	8	9	8	3	3	3	3	8	5	4	2	2	4	5	5	8	8
9	8	8	8	8	9	7	3	7	7	5	7	7	4	4	4	5	4	8	8
10	8	8	9	9	3	3	5	3	5	7	7	7	4	4	4	5	4	7	8
11	8	9	6	3	3	3	5	5	5	5	7	5	4	4	5	5	7	8	8
12	6	6	3	3	3	6	8	7	7	6	6	7	5	4	4	4	5	8	8
13	9	3	3	3	8	7	8	8	8	7	8	9	8	5	7	5	4	8	8
14	3	3	3	8	8	8	8	8	8	8	8	9	9	6	6	5	4	3	7
15	9	8	7	8	8	8	8	8	8	8	8	9	10	7	6	3	2	4	7
16	10	10	7	8	8	8	8	8	8	8	9	9	9	8	7	2	2	4	5
17	10	10	7	6	7	8	8	8	8	8	8	9	8	8	7	4	4	5	5
18	10	10	7	7	7	7	7	8	8	8	8	8	8	8	7	5	4	4	5
19	7	3	7	7	7	7	7	8	8	8	8	8	9	8	7	5	4	4	5
20	7	5	7	7	7	7	8	8	8	8	8	8	9	9	7	5	4	4	5
21	7	5	5	7	7	7	8	8	8	8	8	8	8	9	7	5	4	4	7
22	7	7	5	7	7	7	7	8	8	8	8	8	8	8	7	7	7	7	8

Figura 4.3: Clasificación obtenida mediante trabajo de campo centrada en los 2 píxeles de estudio.

Row	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170
0	8	7	5	5	2	1	2	6	5	2	2	2	4	3	3	3	6	8	7
1	8	7	7	5	4	2	2	3	5	4	4	4	5	4	4	7	6	8	8
2	8	9	8	7	7	5	4	4	5	5	5	7	3	4	3	7	7	8	8
3	9	9	9	9	8	7	7	7	4	5	5	7	3	3	3	7	8	8	8
4	10	9	9	9	8	8	8	8	8	5	3	4	3	9	6	5	8	8	8
5	9	9	9	9	8	8	9	8	3	3	3	6	6	8	7	7	8	8	8
6	8	8	9	9	8	8	6	6	3	3	6	4	7	8	5	5	7	8	8
7	8	8	8	9	8	7	3	3	3	3	7	4	4	2	4	5	7	8	8
8	8	8	8	9	9	8	3	3	3	8	5	4	2	2	4	5	5	8	8
9	8	8	8	8	9	7	3	7	7	5	7	7	4	4	4	5	4	8	8
10	8	8	9	9	3	3	5	3	5	7	7	7	4	4	4	5	4	7	8
11	8	9	6	3	3	3	5	5	5	5	7	5	4	4	5	5	7	8	8
12	6	6	3	3	3	6	8	7	7	6	6	7	5	4	4	4	7	8	8
13	9	3	3	3	8	7	8	8	8	7	8	9	8	5	7	5	4	8	8
14	3	3	3	8	8	8	8	8	8	8	8	9	9	6	6	5	4	3	7
15	9	8	7	8	8	8	8	8	8	8	8	9	10	7	6	3	2	4	7
16	10	10	7	8	8	8	8	8	8	8	9	9	9	8	7	2	2	4	5
17	10	10	7	6	7	8	8	8	8	8	8	9	8	8	7	4	4	5	5
18	10	10	7	7	7	7	7	8	8	8	8	8	8	8	7	5	4	4	5
19	7	3	7	7	7	7	7	8	8	8	8	8	9	8	7	5	4	4	5
20	7	5	7	7	7	7	8	8	8	8	8	8	9	9	7	5	4	4	5
21	7	5	5	7	7	7	8	8	8	8	8	8	8	9	7	5	4	4	7
22	7	7	5	7	7	7	7	8	8	8	8	8	8	8	7	7	7	7	8

Figura 4.4: Clasificación obtenida mediante el algoritmo mínima distancia centrada en los 2 píxeles de estudio.

Vamos a analizar por qué se han clasificado erróneamente dichos píxeles. La Figura 4.5 muestra los valores espectrales de ambos píxeles en las 7 capas de la imagen Landsat, y la Tabla 4.4 resume dichos valores espectrales, que curiosamente son iguales para ambos píxeles con lo que, por simplificación, vamos a tratar ambos píxeles como uno solo.

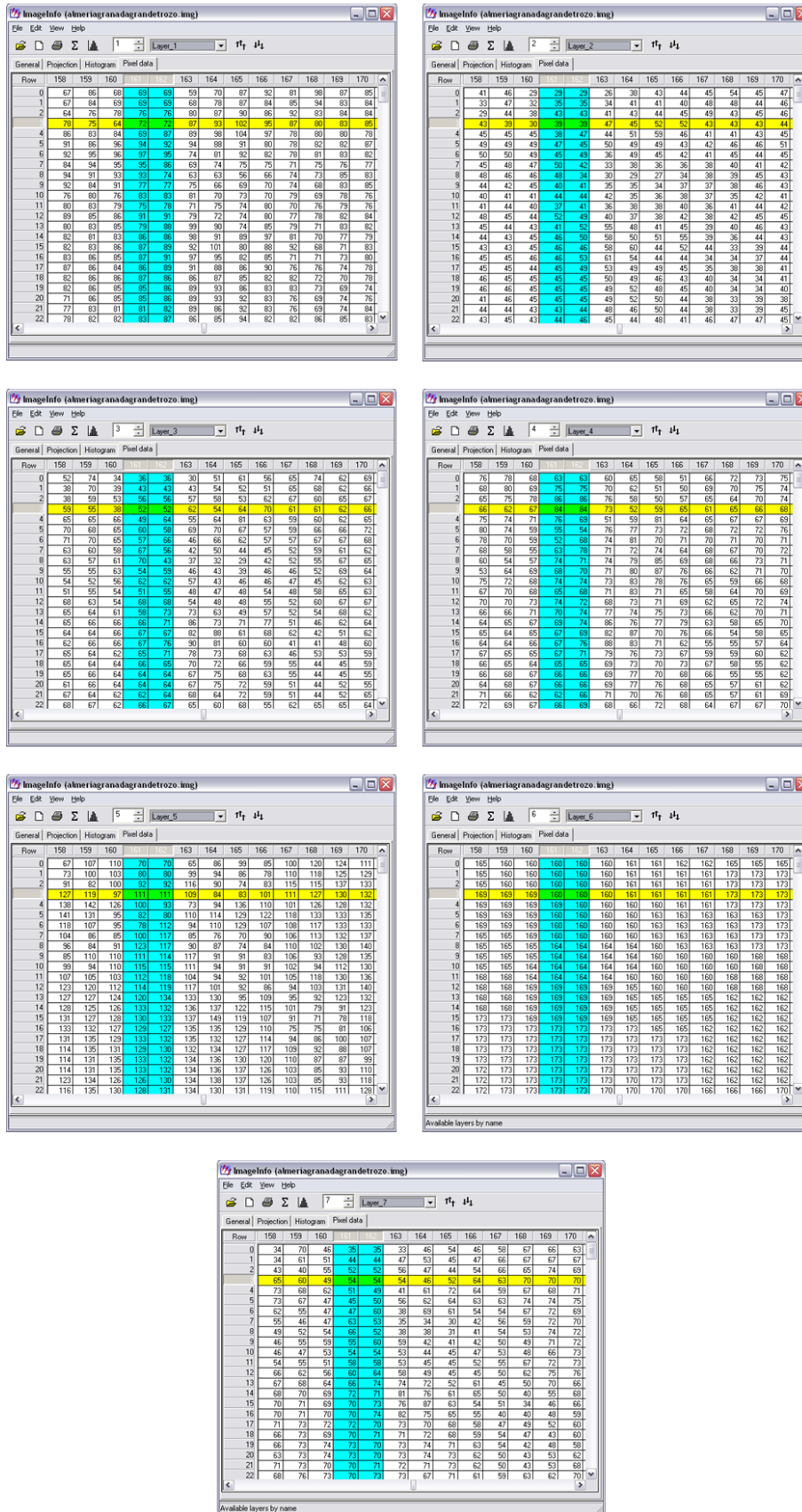


Figura 4.5: Valores espectrales de 2 píxeles mal clasificados en las 7 capas.

Píxel	Capa 1	Capa 2	Capa 3	Capa 4	Capa 5	Capa 6	Capa 7
[3, 161]	72	39	52	84	111	160	54
[3, 162]	72	39	52	84	111	160	54

Tabla 4.4: Valores espectrales de 2 píxeles mal clasificados.

La Tabla 4.5 muestra los valores espectrales de las muestras tomadas para cada una de las 8 clases en las que se ha clasificado la región de estudio, también para cada una de las 7 capas de la imagen Landsat.

Clases	Capa 1	Capa 2	Capa 3	Capa 4	Capa 5	Capa 6	Capa 7
1	56.7658	25.5007	28.0801	48.5983	53.8074	142.911	24.5118
2	65.6057	31.5801	38.1911	53.6608	74.2405	150.921	36.6283
3	91.8313	46.3399	59.8819	62.9785	96.1951	167.733	51.7341
4	73.7848	36.7980	46.8314	56.1121	89.0304	156.462	45.9838
5	76.6276	39.2358	51.7107	59.9246	104.636	160.299	54.5456
6	100.406	53.8322	71.6157	72.6973	114.660	170.330	62.3255
7	81.3447	42.9589	58.5874	64.9728	117.165	165.610	61.9089
8	84.8517	46.2963	65.4349	70.5442	130.463	170.348	69.9688

Tabla 4.5: Valores espectrales de las 8 clases seleccionadas.

Las Tablas 4.6 y 4.7 resumen los valores calculados por el algoritmo de mínima distancia (diferencia al cuadrado de los valores espectrales del píxel y las clases, para cada capa de la imagen).

	Capa 1	Capa 2	Capa 3	Capa 4	Capa 5	Capa 6	Capa 7
<b>Clase 5</b>	76.6276	39.2358	51.7107	59.9246	104.636	160.299	54.5456
<b>Píxel [3, 161]</b>	72	39	52	84	111	160	54
<b>Diferencia<sup>2</sup></b>	21,41	0,06	0,08	579,62	40,50	0,09	0,30

Tabla 4.6: Distancia del píxel erróneo a la clase 5.

	Capa 1	Capa 2	Capa 3	Capa 4	Capa 5	Capa 6	Capa 7
<b>Clase 7</b>	81.3447	42.9589	58.5874	64.9728	117.165	165.610	61.9089
<b>Píxel [3, 161]</b>	72	39	52	84	111	160	54
<b>Diferencia<sup>2</sup></b>	87,32	15,67	43,39	362,03	38,01	31,47	62,55

Tabla 4.7: Distancia del píxel erróneo a la clase 7.

Si sumamos los valores *diferencia*<sup>2</sup> de las tablas anteriores, obtenemos una distancia de 642.06 a la clase 5 y 640.45 a la clase 7. Como se puede observar, la distancia espectral de dicho píxel ambas clases es muy parecida (1.61). Por lo tanto se trata de un píxel incierto, ya que se encuentra muy próximo a ambas clases desde el punto de vista espectral. Se podría utilizar información contextual durante el propio proceso de clasificación para ayudar a este tipo de píxeles, ya que como se vio en las imágenes 4.3 y 4.4, se encontraba rodeado en su mayoría por píxeles de la clase 5, en lugar de píxeles de la clase 7.

Aparte de los píxeles inciertos, también existen los píxeles ruidosos, en los que nos encontramos ruido de tipo impulsivo gaussiano en sus valores espectrales. No son muy frecuentes, pero también pueden ser tratados mediante algoritmos contextuales, en procesos de pre-clasificación de la imagen. La Figura 4.6 muestra una imagen con un 1% de ruido impulsivo gaussiano introducido artificialmente por un programa creado en el proyecto I+D SOLERES, y la Figura 4.7 muestra el valor espectral de un píxel ruidoso.

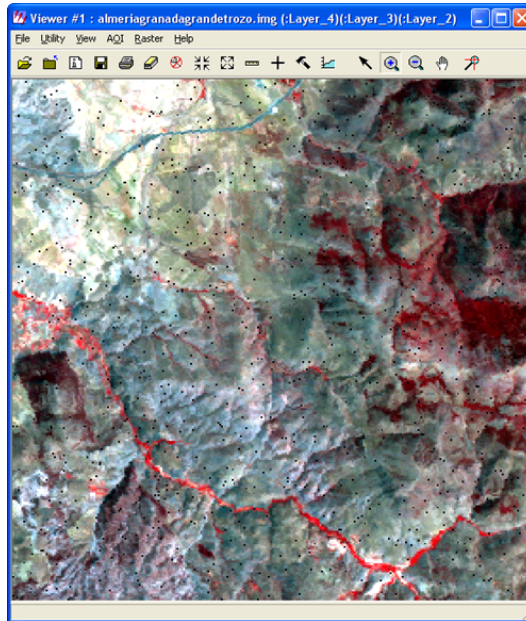


Figura 4.6: Imagen de satélite con 1% de ruido.

Row	0	1	2	3	4	5	6	7	8	9	10	11	12
0	15.000	12.000	16.000	11.000	13.000	10.000	06.000	08.000	01.000	06.000	09.000	08.000	99.0
1	15.000	10.000	10.000	05.000	15.000	14.000	09.000	00.000	03.000	03.000	12.000	06.000	05.0
2	13.000	05.000	07.000	03.000	10.000	11.000	08.000	11.000	06.000	09.000	15.000	98.000	98.0
3	06.000	02.000	04.000	13.000	11.000	09.000	09.000	11.000	10.000	07.000	04.000	85.000	99.0
4	97.000	95.000	99.000	09.000	10.000	09.000	04.000	09.000	97.000	96.000	94.000	00.000	97.0
5	94.000	95.000	00.000	12.000	14.000	11.000	09.000	10.000	02.000	83.000	85.000	94.000	97.0
6	98.000	98.000	05.000	15.000	17.000	14.000	08.000	14.000	98.000	89.000	91.000	89.000	92.0
7	98.000	05.000	17.000	13.000	14.000	22.000	10.000	01.000	87.000	85.000	96.000	00.000	98.0
8	03.000	03.000	11.000	07.000	09.000	99.000	94.000	93.000	94.000	01.000	03.000	02.000	10.0
9	15.000	12.000	12.000	11.000	97.000	95.000	99.000	98.000	94.000	92.000	00.000	13.000	07.0
10	11.000	18.000	18.000	16.000	01.000	09.000	03.000	01.000	99.000	06.000	08.000	14.000	18.0
11	03.000	18.000	15.000	18.000	05.000	02.000	03.000	90.000	00.000	06.000	08.000	09.000	07.0
12	16.000	22.000	12.000	15.000	96.000	96.000	92.000	01.000	17.000	10.000	06.000	10.000	02.0
13	18.000	19.000	18.000	20.000	07.000	12.000	04.000	15.000	23.000	09.000	08.000	14.000	88.0
14	27.000	37.000	34.000	11.000	18.000	18.000	17.000	21.000	19.000	02.000	04.000	08.000	90.0
15	17.000	44.000	27.000	12.000	21.000	24.000	30.000	20.000	98.000	97.000	07.000	15.000	12.0
16	08.000	24.000	09.000	09.000	11.000	05.000	09.000	99.000	11.000	11.000	22.000	14.000	23.0
17	08.000	08.000	07.000	07.000	11.000	15.000	05.000	05.000	05.000	08.000	12.000	14.000	28.0
18	96.000	03.000	10.000	05.000	07.000	09.000	12.000	12.000	11.000	07.000	10.000	07.000	15.0
19	05.000	15.000	13.000	00.000	99.000	05.000	03.000	03.000	07.000	03.000	01.000	09.000	02.0

Figura 4.7: Valor espectral de un píxel ruidoso.

Estos algoritmos contextuales de pre-clasificación (para píxeles ruidosos) y post-clasificación (para píxeles inciertos) pueden ser implementados mediante el uso de autómatas celulares usando la propiedad de vecindad y optimizados mediante la ejecución en distintas iteraciones, como se muestra en el siguiente apartado.

## 4.2. CLASIFICACIÓN DE IMÁGENES DE SATÉLITE CON AUTÓMATAS CELULARES

En este apartado se citan el trabajo previo existente relacionado con la clasificación de imágenes con autómatas celulares, se enumeran los objetivos que pretende alcanzar el algoritmo ACA y se muestran las reglas y estados de la versión ACA 1.0.

### 4.2.1. Trabajo previo existente

A pesar del gran número de clasificadores disponibles en la actualidad, existen muchos grupos de investigación tratando de estudiar nuevos métodos de clasificación de imágenes de satélite, ya que todavía no se ha creado un clasificador que ofrezca una eficiencia del 100 %. En este trabajo se propone un nuevo procedimiento para la clasificación de imágenes de satélite. El nuevo algoritmo de clasificación, basado en autómatas celulares llamado ACA (Algorithm based on Cellular Automata), utiliza esta técnica para la asignación de los píxeles de la imagen a las distintas clases espectrales.

Hasta ahora, los autómatas celulares han sido aplicados principalmente para implementar todo tipo de simulaciones [8]. En el ámbito de la teledetección, este tipo de simulaciones han sido sobre todo de aspecto medioambiental y social. En este trabajo se propone una alternativa importante y novedosa: la aplicación de autómatas celulares en imágenes de satélite para implementar un algoritmo de clasificación contextual supervisado.

La aplicación de autómatas celulares en los procesos de clasificación de imágenes de satélite es un nuevo campo de estudio en el que apenas hay investigaciones significativas. No hay apenas trabajos previos relativos a la clasificación de imágenes de satélite mediante autómatas celulares, tan sólo un trabajo sobre autómatas celulares aplicados en procesos de post-clasificación para mejorar el resultado obtenido previamente con otro algoritmo de clasificación [41] y otro en el que se utilizan los autómatas celulares combinados con algoritmos genéticos para el reconocimiento de patrones [37]. En este artículo se propone una metodología para implementar un nuevo algoritmo de clasificación de imágenes satelitales mediante autómatas celulares (algoritmo ACA), que clasifica a los píxeles de la imagen basándose tanto en información espectral como contextual de cada píxel, y por lo tanto mejora los resultados obtenidos por otros algoritmos de clasificación existentes en la literatura.

Este nuevo proceso de clasificación se ha incorporado en un Sistema de Información Ambiental desarrollado para el marco de estudio del proyecto I+D SOLERES. Este tipo de sistema de información tiene como objetivo solucionar problemas medioambientales, facilitando el análisis de riesgos y la mejora de la gestión ambiental.

El algoritmo ACA se ha implementado en Visual C++ con la ayuda del Toolkit de la herramienta Erdas Imagine 9.1, y es un algoritmo de clasificación basado en un autómata celular bidimensional y multiestado que ofrece al usuario la posibilidad realizar una clasificación lo más personalizada posible, ya que le permite introducir nuevos estados y reglas para ajustar el algoritmo a un interés concreto y específico.

### 4.2.2. Objetivos principales del algoritmo ACA

En esta sección se detallan los principales objetivos que se pretenden alcanzar a través de un algoritmo de clasificación basado en autómatas celulares. Para aplicar el algoritmo ACA a una imagen de satélite, debemos tener en cuenta previamente las siguientes correspondencias entre el autómata celular y los elementos básicos de un proceso genérico de clasificación de imágenes de satélite:

- Cada celda del autómata celular corresponde a un píxel de la imagen.
- Los estados de cada celda nos indican la clase a la que pertenece cada píxel.
- La vecindad de cada celda puede estar formada por 4 píxeles (vecindad von Neumann), 8 píxeles (vecindad de Moore) o 24 píxeles (vecindad de Moore extendida).
- La función de transición  $f$  debe clasificar cada píxel de la imagen basándose en las características de la celda actual y de su vecindad, y aplicando las reglas establecidas a lo largo de las iteraciones del proceso de clasificación.

Con el fin de personalizar el proceso de clasificación, el analista experto de imágenes de satélite debe establecer el comportamiento deseado del algoritmo ACA a través de la definición de los estados y reglas correspondientes del autómata celular que ajuste el proceso de clasificación para obtener los resultados deseados. Por ejemplo, en este trabajo se ha implementado una primera versión del algoritmo ACA (al que se ha llamado ACA v1.0) que tiene como finalidad conseguir 3 objetivos distintos:

- *Objetivo 1*: mejorar los resultados obtenidos por los algoritmos clásicos de clasificación supervisada a través del uso de información contextual, para no clasificar erróneamente los píxeles inciertos y ruidosos. Para ello, el algoritmo ACA clasificará los píxeles problemáticos teniendo en cuenta no sólo sus valores espectrales (ambiguos en el caso de los píxeles inciertos e erróneos en el caso de los píxeles ruidosos), sino también los valores de los vecinos para afinar más en el resultado final de la clasificación. Gracias a este objetivo, se consigue fusionar en un solo algoritmo tres procesos que normalmente se utilizan por separado y en el siguiente orden: pre-clasificación (eliminación de píxeles ruidosos), clasificación propiamente dicha y post-clasificación (afinamiento de píxeles inciertos).
- *Objetivo 2*: obtener una clasificación jerarquizada dividida en capas de fiabilidad, donde en cada iteración del autómata celular son clasificados sólo aquellos píxeles que están dentro de una distancia espectral máxima en el espacio de características respecto del centro de su clase correspondiente, y esta distancia se incrementa en cada iteración. De este modo obtenemos una clasificación dividida en jerarquías de calidad, donde las primeras capas ofrecen mayor fiabilidad que las últimas, información que puede resultar muy útil en el posterior análisis de los resultados por parte de los expertos. Además, este proceso ayuda al primer objetivo ya que los píxeles inciertos y ruidosos, que están más alejados normalmente del centro de sus clases, son clasificados en las últimas iteraciones del autómata celular, utilizando como vecinos a píxeles que con una gran probabilidad se han clasificado en iteraciones anteriores, y por lo tanto ofrecen mayor fiabilidad, mejorando así el índice de la calidad global del proceso de clasificación optimizado.
- *Objetivo 3*: conseguir una lista detallada de los píxeles inciertos y ruidosos, algo que puede ser útil por si fallara el proceso de clasificación aún usando técnicas contextuales, y una lista de los píxeles que determinan los bordes espaciales de las clases de la imagen. Esta segunda opción es bastante interesante, ya que la detección de bordes espaciales (no espectrales) de las clases también es un problema dentro del ámbito de la teledetección. Gracias a esta información, podemos establecer por ejemplo en futuros análisis cuántas agrupaciones de una misma clase existen en una imagen de satélite determinada (por ejemplo, contar el número de edificios de una ciudad).

### 4.2.3. Algoritmo ACA v1.0: estados y reglas

Para conseguir los objetivos citados en la sección anterior se ha desarrollado un autómata celular multiestado en el que cada celda tiene 3 estados independientes, donde cada estado cubre uno de los 3 objetivos. El primer estado es [*clase*], y corresponde a la clase en la que es clasificado cada píxel de la imagen utilizando no sólo sus valores espectrales, sino también información contextual. Este estado nos permite mejorar el índice de calidad de la clasificación (objetivo 1 de ACA v.1.0). El segundo estado es [*calidad*], y corresponde al número de iteración del autómata celular en el que es clasificado cada píxel de la imagen. Este estado nos permite obtener la clasificación jerarquizada basada en niveles de fiabilidad (objetivo 2 de ACA v.1.0). El tercer estado es [*tipo*], y corresponde al tipo de píxel con el que estamos tratando: incierto, ruidoso, borde o foco (los que no son bordes espaciales de las clases). Este estado nos permite obtener una lista detallada de los píxeles inciertos, ruidosos y bordes de clases (objetivo 3 de ACA v.1.0). A continuación se muestran los valores que puede tomar cada uno de los 3 estados del autómata celular:

- *Estado 1* [*clase*] = *claseEspectral* (clases definidas por el conjunto de entrenamiento), *claseRuido* (píxeles ruidosos) o *claseVacía* (píxeles que todavía no han sido clasificados).
- *Estado 2* [*calidad*] =  $1 \dots numIteraciones$  (iteración del autómata celular que indica la capa de fiabilidad de la clasificación jerárquica).
- *Estado 3* [*tipo*] = *incierto* (píxeles dudosos), *ruidoso* (píxeles ruidosos), *borde* (píxeles frontera de las clases) y *foco* (píxeles que no son bordes).

El estado 1 [*clase*] puede tomar cualquier clase espectral que se haya definido anteriormente en el conjunto de entrenamiento, así como la clase ruido o clase vacía para aquellos píxeles que aún no han sido clasificados (en la primera iteración del autómata celular, todas las celdas tienen su estado 1 con el valor clase vacía). El estado 2 [*calidad*] toma el valor de la iteración en la que dicho píxel es clasificado, un valor definido entre 1 y el número máximo de iteraciones. El estado 3 [*tipo*] puede tomar el tipo de píxel con el que nos encontramos: incierto, ruidoso, borde o foco.

Esta versión del algoritmo ACA está basado en el clasificador mínima distancia supervisado, por lo que el autómata celular utiliza los resultados de la clasificación espectral mínima distancia para posteriormente aplicar las reglas de su función de transición  $f$ . En cada iteración del autómata celular, se va aumentando la distancia espectral permitida, o el radio de pertenencia posible a las clases en el espacio de características, de tal modo que en las primeras iteraciones se clasifican los píxeles cuya distancia euclídea a su clase correspondiente (algoritmo mínima distancia) sea muy baja, y en las siguientes iteraciones se va aumentando el radio de búsqueda espectral. De este modo conseguimos realizar una clasificación jerarquizada basada en capas de fiabilidad. La función de transición  $f$  debe tener en cuenta las siguientes entradas:

- Los estados vecinos: los estados de la vecindad del píxel actual. Esta vecindad puede ser de tipo von Neumann, Moore o Moore extendida.
- Clasificación espectral obtenida mediante el algoritmo modificado de mínima distancia: clases a las que puede pertenecer el píxel actual (una o varias clases, si es un píxel incierto habrá dudas entre varias clases).
- Iteración del AC: la iteración actual de la autómata celular.



Para conseguir los tres objetivos citados anteriormente se han creado cuatro reglas que definen el comportamiento del autómata celular multiestado, una para cada tipo de píxel especificado en el tercer objetivo del algoritmo ACA:

- *Regla 1*: si el número de clases posibles obtenidas en la clasificación espectral es 1, y las clases de la vecindad son *clase Vacía* o iguales a la del píxel actual:

$$[clase] [calidad] [tipo] = [claseEspectral] [iteracionAC] [foco]$$

- *Regla 2*: si el número de clases posibles obtenidas en la clasificación espectral es 1, y las clases de la vecindad son distintas a la del píxel actual:

$$[clase] [calidad] [tipo] = [claseEspectral] [iteracionAC] [borde]$$

- *Regla 3*: si el número de clases posibles obtenidas en la clasificación espectral es 1, y *claseEspectral* es *claseRuido*:

$$[clase] [calidad] [tipo] = [clase\ mayoría\ de\ la\ vecindad] [iteracionAC] [ruidoso]$$

- *Regla 4*: si el número de clases posibles obtenidas en la clasificación espectral es mayor que 1:

$$[clase] [calidad] [tipo] = [clase\ mayoría\ de\ la\ vecindad\ de\ entre\ las\ clases\ dudosas] [iteracionAC] [incierto]$$

Como se puede observar, cada regla corresponde a uno de los tipos de píxel que se han especificado previamente: la regla 1 para el píxel tipo foco, la regla 2 para el píxel tipo borde, la regla 3 para el píxel tipo ruidoso y la regla 4 para el píxel tipo incierto. En las tres primeras reglas, la primera condición es la misma: que el número de clases posibles obtenidas en la clasificación espectral (es decir, mediante el algoritmo mínima distancia modificado) sea igual a 1. Es decir, son píxeles que sabemos a ciencia cierta a qué clase pertenece, o como ocurre en el tercer caso (el de los píxeles ruidosos), al menos sabemos que se trata de un píxel de ese tipo. Esta condición cambia en el cuarto tipo de píxel, ya que los píxeles inciertos se caracterizan precisamente por poder pertenecer a varias clases, si nos basamos sólo en sus valores espectrales y no tomamos como información suplementaria las clases de sus píxeles vecinos. A continuación se realiza una explicación más detallada del funcionamiento de cada una de las reglas:

- *Regla 1*: un píxel será etiquetado como foco si no es el borde espacial de una clase, por lo tanto si los píxeles de alrededor pertenecen a la misma clase o a la clase vacía (aún no han sido clasificados).
- *Regla 2*: un píxel será etiquetado como borde si es la frontera espacial de una clase, por lo tanto existen píxeles vecinos que pertenecen a una clase distinta de la clase de dicho píxel.
- *Regla 3*: un píxel será etiquetado como ruidoso si sus valores espectrales son erróneos (todas las capas a valor 0), y por lo tanto tomará como clase la clase mayoritaria de los píxeles vecinos.
- *Regla 4*: un píxel será etiquetado como incierto si su clasificación espectral es mayor de una sola clase, y tomara como clase la clase mayoritaria de los píxeles vecinos de entre las dos o más clases entre las que se duda.

### 4.3. DESCRIPCIÓN DEL ALGORITMO ACA

En este apartado se explica en profundidad la arquitectura completa del algoritmo ACA, se expone el pseudocódigo del algoritmo y se muestra una simulación del mismo.

#### 4.3.1. Arquitectura del algoritmo ACA

Con el objetivo de crear un algoritmo de clasificación de imágenes de satélite basado en el comportamiento de los autómatas celulares se ha creado una arquitectura software completa (llamada arquitectura ACA) con distintas aplicaciones complementarias que interactúan entre sí. La Figura 4.8 muestra la arquitectura ACA completa.

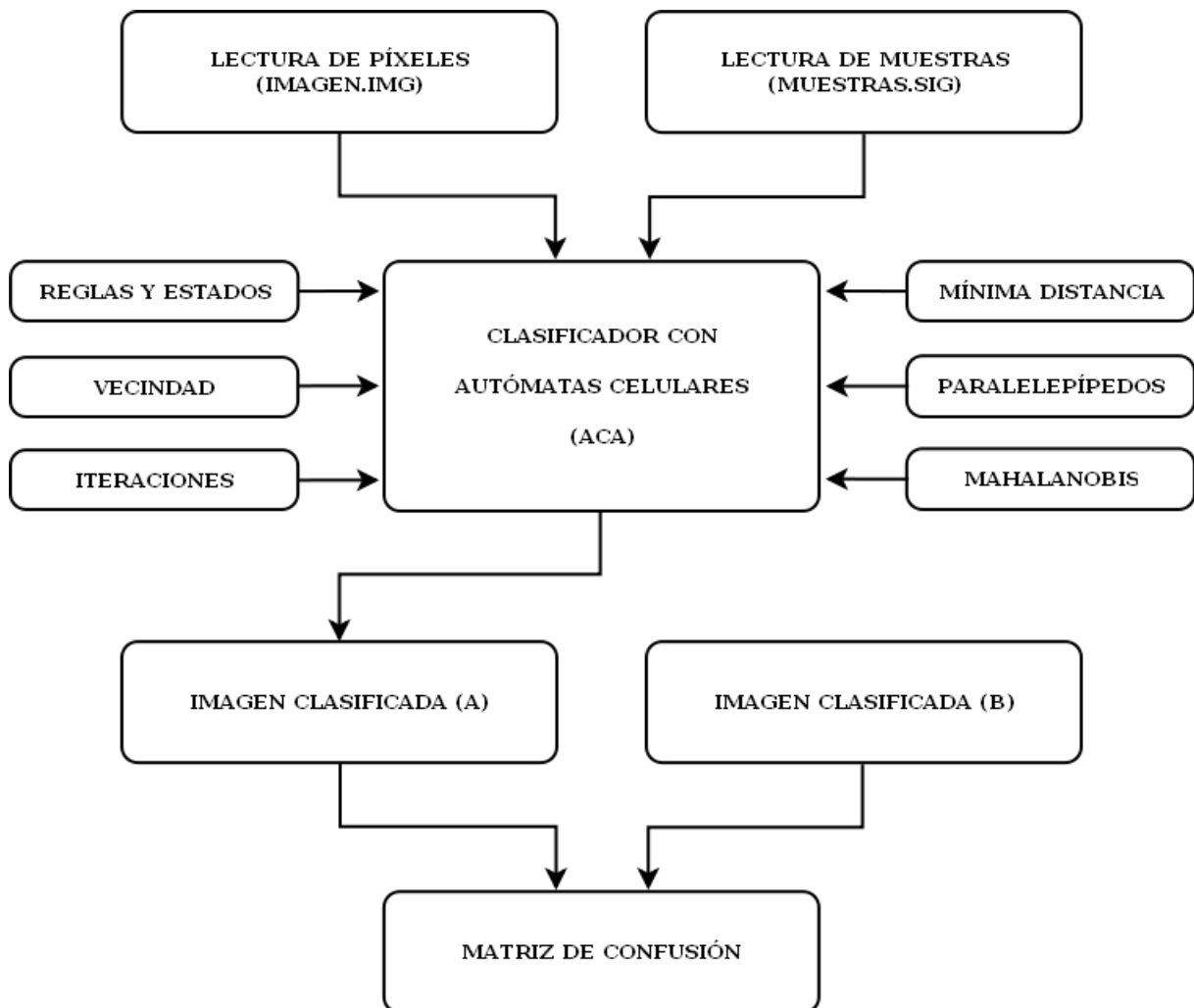


Figura 4.8: Arquitectura completa del clasificador ACA.

La arquitectura ACA está compuesta de dos algoritmos principales: el algoritmo ACA propiamente dicho y la aplicación del cálculo de calidad de la clasificación obtenida. A continuación se explica el funcionamiento de ambos algoritmos principales:

1. Algoritmo ACA. El algoritmo ACA tiene ocho componentes de entrada y una sola salida: la imagen clasificada. Los ocho componentes de entrada se pueden agrupar en tres categorías distintas:

- Componentes de entrada del algoritmo ACA relacionados con la imagen a clasificar. El algoritmo ACA tiene dos entradas de este tipo: la aplicación de lectura de la imagen de satélite (*imagen.img*) y la aplicación de lectura del conjunto de muestras (*muestras.sig*) obtenido previamente a partir de píxeles que pertenecen a las clases elegidas para el proceso de clasificación. Con estos dos elementos se prepara al algoritmo ACA para realizar una clasificación de tipo supervisada.
  - Componentes de entrada del algoritmo ACA relacionados con algoritmos clásicos de clasificación supervisada. El algoritmo ACA está basado en los resultados obtenidos previamente mediante algún algoritmo de clasificación supervisado clásico modificado. Concretamente, se han elegido los clasificadores supervisados de mínima distancia, paralelepípedos y máxima verosimilitud. Estos algoritmos utilizan los datos introducidos al algoritmo ACA (*imagen.img* y *muestras.sig*) para clasificar la imagen, pero no utilizando los algoritmos clásicos, sino modificados y mejorados mediante el uso de autómatas celulares. El usuario elige, antes de llevar a cabo el proceso de clasificación, el algoritmo supervisado clásico en el que quiere que se base el nuevo algoritmo ACA.
  - Componentes de entrada del algoritmo ACA relacionados con autómatas celulares. Para mejorar los resultados obtenidos mediante los algoritmos clásicos supervisados, el algoritmo ACA modifica su comportamiento mediante el uso de autómatas celulares, añadiendo los elementos de estados, reglas, vecindad e iteraciones al proceso de clasificación supervisado. Todos estos componentes relacionados con autómatas celulares pueden ser configurados por el usuario antes de llevar a cabo el proceso de clasificación. De este modo, el usuario puede definir nuevos estados y reglas para personalizar más los resultados finales de clasificación; también puede cambiar la vecindad (vecindad de von Neumann, vecindad de Moore o vecindad de Moore extendida), así como el número de iteraciones, configurando el algoritmo ACA de tal modo que se optimicen al máximo los resultados obtenidos.
2. Aplicación del cálculo de calidad de la clasificación obtenida. Esta aplicación toma como entradas la imagen clasificada mediante el algoritmo ACA (*imagen A*) y la imagen clasificada mediante trabajo de campo (*imagen B*), y obtiene la matriz de confusión de ambas imágenes, ofreciendo como salida un índice que indica el grado de acierto en el proceso de clasificación, es decir, el porcentaje de píxeles bien clasificados. Además, ofrece una lista de los píxeles mal clasificados, relacionando la clase a la que pertenece realmente (trabajo de campo) con la clase en la que ha sido clasificado (clasificación ACA), y de este modo se puede determinar si existe confusión entre clases por estar muy próximas entre sí desde el punto de vista espectral.

### 4.3.2. Pseudocódigo del algoritmo ACA

Esta sección explica el algoritmo en pseudocódigo corresponde a la versión ACA v1.0 descrita en apartados anteriores, que se ha dividido en 3 partes distintas: ACA principal, ACA espectral y ACA contextual. A continuación se explica cada algoritmo:

- Algoritmo ACA principal: ejecuta las iteraciones del autómata celular. En cada iteración, primero se realiza una clasificación espectral de todos los píxeles que aún no han sido clasificados en la imagen, y posteriormente una clasificación contextual de aquellos que se encuentran en el radio de pertenencia a su clase permitido. La Tabla 4.8 muestra el pseudocódigo del algoritmo ACA principal.

- Algoritmo ACA espectral: está basado en algoritmos clásicos supervisados mejorados mediante técnicas de autómatas celulares gracias a la división del proceso de clasificación en varias iteraciones, lo que provoca que se obtenga una clasificación jerarquizada en distintas capas con un nivel de fiabilidad cada una. Es decir, los píxeles clasificados en las primeras iteraciones son más fiables que los clasificados en las siguientes iteraciones. En esta parte del algoritmo, por lo tanto, se consigue el objetivo 2 de ACA v1.0. La Tabla 4.9 muestra el pseudocódigo del algoritmo ACA espectral.
- Algoritmo ACA contextual: utiliza la vecindad del autómata celular para aplicar una clasificación contextual, así como un conjunto de estados y reglas para personalizar los resultados obtenidos del proceso de clasificación. Este algoritmo se basa en los resultados obtenidos mediante el algoritmo ACA espectral, que son mejorados mediante técnicas contextuales y personalizados mediante el uso de reglas y estados del autómata celular. En esta parte del algoritmo se consiguen los otros dos objetivos planteados. Por un lado, se consigue mejorar la calidad de la clasificación obtenida mediante técnicas contextuales (objetivo 1 de ACA v1.0), concretamente aplicando las reglas 3 y 4 definidas con anterioridad (para píxeles ruidosos e inciertos). Por otro lado, también se consigue definir el tipo de píxel en el siguiente orden: foco, borde, ruidoso o incierto, aplicando las 4 reglas del autómata celular definidas en secciones anteriores (objetivo 3 de ACA v1.0). La Tabla 4.10 muestra el pseudocódigo del algoritmo ACA contextual.

---

**Algoritmo ACA principal** ( $E, numClasses, numIterations, threshold$ )

---

**Entrada:**

$E = \{e_1, e_2, \dots, e_n\}$ : conjunto de píxeles a clasificar

$numClasses$ : número de clases

$numIterations$ : máximo de iteraciones del AC

$threshold$ : incremento en cada iteración del AC

**Salida:**

$C = \{c_1, c_2, \dots, c_k\}$ : conjunto de centros de las clases

$L = \{l(e) \mid e = 1, 2, \dots, n\}$ : píxeles clasificados

---

```

01   for  $i \leftarrow 0$  to  $numIterations$  do
02       foreach  $e_i \in E$  do
03           if  $e_i.classificated \neq true$  then
04                $spectralClasses \leftarrow spectralACA(e_i, numClasses, threshold)$ ;
05               if  $spectralClasses \neq \emptyset$  then
06                    $finalClass \leftarrow contextualACA(e_i, spectralClasses)$ ;
07               endif
08           endif
09       end
10        $threshold \leftarrow threshold + incremental$ ;
11   end

```

---

Tabla 4.8: Algoritmo ACA principal.

---

**Algoritmo ACA espectral** ( $e_i, numClasses, threshold$ )

---

**Entrada:**  $e_i$ : píxel a clasificar,  $numClasses$ : número de clases,  
 $threshold$ : incremento en cada iteración del AC

**Salida:**  $spectralClasses$ : clases espectrales a las que puede pertenecer el píxel

---

```

01   $spectralClasses \leftarrow \emptyset$ ;
02  for  $j \leftarrow 0$  to  $numClasses$  do
03      if  $minDistance(e_i, class_j) \mid j \in \{1..k\} \leq threshold$  then
04           $spectralClasses \leftarrow spectralClasses \cup class_j$ ;
05      endif
06  end

```

---

Tabla 4.9: Algoritmo ACA espectral.

---

**Algoritmo ACA contextual** ( $e_i, spectralClasses$ )

---

**Entrada:**  $e_i$ : píxel a clasificar,  
 $spectralClasses$ : clases espectrales a las que puede pertenecer el píxel

**Salida:**  $finalClass$ : clase final del píxel

---

```

01  if  $size(spectralClasses) = 1$  then
02      if  $\{spectralClasses\} \neq noiseClass$  then
03           $e_i.class \leftarrow \{spectralClasses\}$ ;
04           $e_i.quality \leftarrow numIteration$ ;
05          if  $neighbourhoodClassesType(e_i) = 1$  then
06               $e_i.type \leftarrow focusPixel$ ;
07          endif
08          if  $neighbourhoodClassesType(e_i) = 2$  then
09               $e_i.type \leftarrow edgePixel$ ;
10          endif
11      endif
12      if  $\{spectralClasses\} = noiseClass$  then
13           $e_i.class \leftarrow bayesNeighbourhood()$ ;
14           $e_i.quality \leftarrow numIteration$ ;
15           $e_i.type \leftarrow noisePixel$ ;
16      endif
17  endif
18  if  $size(spectralClasses) \neq 1$  then
19       $e_i.class \leftarrow bayesNeighbourhoodClass()$ ;
20       $e_i.quality \leftarrow numIteration$ ;
21       $e_i.type \leftarrow uncertainPixel$ ;
22  endif

```

---

Tabla 4.10: Algoritmo ACA contextual.

### 4.3.3. Simulación del algoritmo ACA

La Figura 4.9 simula el proceso de clasificación expandida jerarquizada del algoritmo ACA mediante 3 iteraciones para los algoritmos mínima distancia, paralelepípedos y máxima verosimilitud. Este proceso se representa en dicha figura en el plano espectral (espacio de características). Como se puede observar, en cada iteración va aumentando el rango de píxeles que se clasifican, ya que aumenta la distancia de pertenencia permitida para cada clase. Los píxeles más próximos espectralmente a las clases son clasificados en la primera iteración (la mayoría pertenecen incluso al conjunto de muestras elegido). En las siguientes iteraciones se clasifican los píxeles más alejados, y por último los píxeles inciertos mediante técnicas de clasificación contextual, con la seguridad de que las técnicas contextuales van a basarse en píxeles vecinos bien clasificados en iteraciones anteriores.

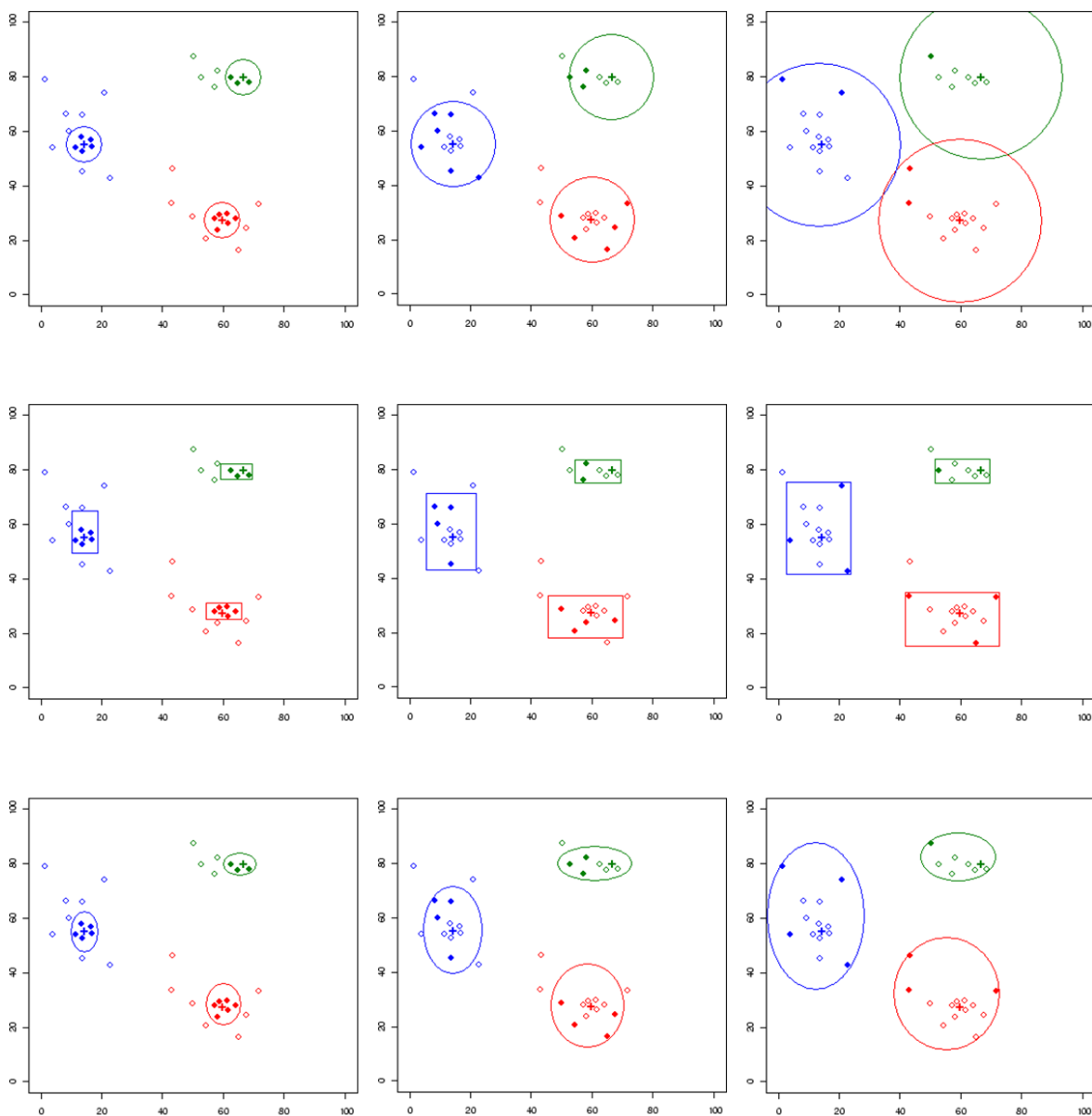


Figura 4.9: Evolución del algoritmo ACA con 3 iteraciones: (primera línea) mínima distancia, (segunda línea) paralelepípedos, (tercera línea) máxima verosimilitud.

#### 4.4. RESUMEN Y CONCLUSIONES DEL CAPÍTULO

El algoritmo ACA utiliza los autómatas celulares no sólo para mejorar la calidad de la clasificación espectral mediante el uso de técnicas contextuales aplicadas a través de la vecindad del autómata celular, sino para obtener una clasificación iterativa en la que los píxeles clasificados en una iteración determinada son más fiables que los píxeles clasificados en la siguiente iteración, y así sucesivamente. Esto permite obtener una clasificación jerarquizada en capas o niveles de fiabilidad, donde cada capa corresponde con una iteración determinada del autómata celular. Los resultados del proceso de clasificación, agrupados de este modo, pueden ser de mucha utilidad para ayudar en el posterior proceso de interpretación de los resultados obtenidos en el proceso de clasificación por parte de los expertos analistas.

Además ofrece una lista de los píxeles inciertos, ruidosos y bordes espaciales de las clases de la imagen de satélite, lo que complementa más aún la información obtenida a través del proceso de clasificación con autómatas celulares.

El siguiente capítulo muestra los resultados obtenidos con el algoritmo ACA v1.0, resultados en los que se ha conseguido alcanzar los tres objetivos propuestos.





---

## CAPÍTULO 5

### RESULTADOS OBTENIDOS

---



# Capítulo 5

## RESULTADOS OBTENIDOS

### Contenidos

---

<b>5.1.</b>	MEJORA DE LA CALIDAD DE LA CLASIFICACIÓN . . . . .	<b>90</b>
<b>5.2.</b>	OBTENCIÓN DE UNA CLASIFICACIÓN JERÁRQUICA . . . . .	<b>91</b>
<b>5.3.</b>	DETECCIÓN DE BORDES, PÍXELES INCIERTOS Y RUIDOSOS . . . . .	<b>95</b>
<b>5.4.</b>	RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .	<b>96</b>

---



**E**l algoritmo de clasificación de imágenes de satélite basado en autómatas celulares ACA ha sido implementado y probado dentro del proyecto I+D SOLERES. Los experimentos se han realizado sobre una imagen Landsat multispectral de 7 bandas, con una resolución de 301x301 píxeles (90.601 píxeles totales). La resolución espacial de cada píxel es de 30mx30m. La imagen corresponde a una región de las provincias de Almería y Granada (España), y se trata de una región geográfica con un porcentaje significativo de píxeles inciertos debido a la heterogeneidad del terreno y una imagen con un porcentaje mínimo de píxeles ruidosos (se le ha añadido artificialmente un 1 % de ruido). La Figura 5.1 muestra la imagen de las provincias de Almería y Granada completas, así como la región tomada para realizar las pruebas del algoritmo ACA.

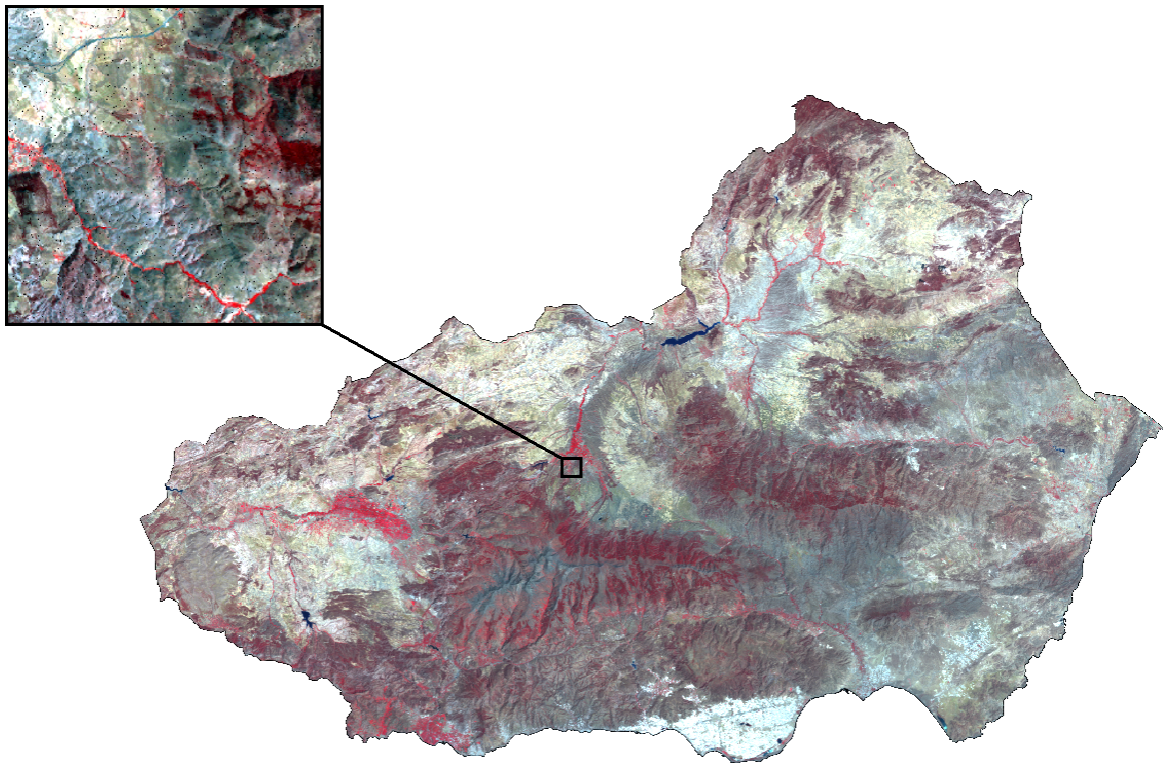


Figura 5.1: Región tomada de la imagen de satélite completa de las provincias de Almería y Granada (España).

Antes de llevar a cabo el proceso de clasificación con autómatas celulares, se ha escogido un conjunto de muestras de cada una de las clases en las que está clasificada la región, según el trabajo de campo previo realizado por el grupo de ecólogos del proyecto I+D SOLERES. El número de clases a los que puede pertenecer cada píxel se ha establecido en 8. Una vez conseguidos los requisitos necesarios para realizar la clasificación supervisada ACA, se ha llevado a cabo el proceso de clasificación, y se han conseguido alcanzar los tres objetivos que se exponen en los siguientes apartados.

## 5.1. MEJORA DE LA CALIDAD DE LA CLASIFICACIÓN

El algoritmo ACA mejora los resultados obtenidos por otros algoritmos de clasificación supervisada, ya que en el proceso de clasificación de cada píxel de la imagen se utilizan los píxeles de alrededor como vecindad de la función de transición  $f$ , y esta relación entre los píxeles de la imagen ofrece una clasificación contextual optimizada. En este experimento se han comparado los resultados de la clasificación obtenida con el algoritmo clásico de mínima distancia y la clasificación del algoritmo ACA basado en mínima distancia. Para comparar ambos algoritmos se ha calculado la matriz de confusión entre las dos imágenes clasificadas y una imagen bien clasificada (obtenida con trabajo de campo). La Tabla 5.1 muestra la matriz de confusión del algoritmo mínima distancia, y la Tabla 5.2 muestra la matriz de confusión del algoritmo ACA.

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8
Clase 1	<b>0</b>	0	0	0	0	0	0	0
Clase 2	789	<b>6080</b>	355	0	0	0	0	0
Clase 3	1247	0	<b>9447</b>	532	0	0	0	0
Clase 4	1547	0	2	<b>11998</b>	242	0	0	0
Clase 5	1555	0	0	52	<b>12827</b>	3	27	0
Clase 6	1027	0	0	47	281	<b>8330</b>	35	1
Clase 7	1513	0	0	0	250	12	<b>13050</b>	0
Clase 8	1396	0	0	0	0	66	381	<b>11242</b>

Tabla 5.1: Matriz de confusión del algoritmo mínima distancia.

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8
Clase 1	<b>0</b>	0	0	0	0	0	0	0
Clase 2	123	<b>6326</b>	611	54	62	11	18	14
Clase 3	179	0	<b>9648</b>	963	240	74	73	37
Clase 4	205	0	5	<b>12282</b>	777	252	174	73
Clase 5	194	0	1	58	<b>13254</b>	88	713	135
Clase 6	123	0	0	47	291	<b>8699</b>	161	330
Clase 7	130	0	0	1	254	16	<b>13724</b>	661
Clase 8	217	0	0	0	0	68	407	<b>11966</b>

Tabla 5.2: Matriz de confusión del algoritmo ACA.

El número de píxeles bien clasificados en cada algoritmo se obtiene sumando los valores de la diagonal principal de la tabla. En el caso del algoritmo de mínima distancia hay un total de 72,974 píxeles bien clasificados (80 % bien clasificados) y, en el algoritmo ACA hay un total de 75,899 píxeles bien clasificados (84 % bien clasificados). Por lo tanto la calidad de la clasificación final ha mejorado en un 4 %.

Además se ha agrupado en el algoritmo ACA el proceso de pre-clasificación (reducción de ruido), proceso de clasificación propiamente dicho y el proceso de post-clasificación (mejora de píxeles inciertos).

## 5.2. OBTENCIÓN DE UNA CLASIFICACIÓN JERÁRQUICA

Con el algoritmo ACA podemos obtener una clasificación jerárquica dividida en capas de fiabilidad basada en la proximidad espectral dentro del espacio de características, de modo que en cada iteración del autómata celular sólo aquellos píxeles de la imagen que están a una distancia máxima del centro de su clase correspondiente son clasificados, y dicha distancia va aumentando en cada iteración. Por lo tanto, los píxeles clasificados en una iteración particular son más fiables que los que se clasifican en una iteración posterior, y así sucesivamente. Este método de clasificación tiene un comportamiento similar a la clasificación difusa, aunque no exactamente igual, por lo que le hemos llamado pseudo-difuso. La Figura 5.2 muestra la secuencia de imágenes obtenida al dividir el proceso de clasificación en 6 iteraciones del autómata celular.

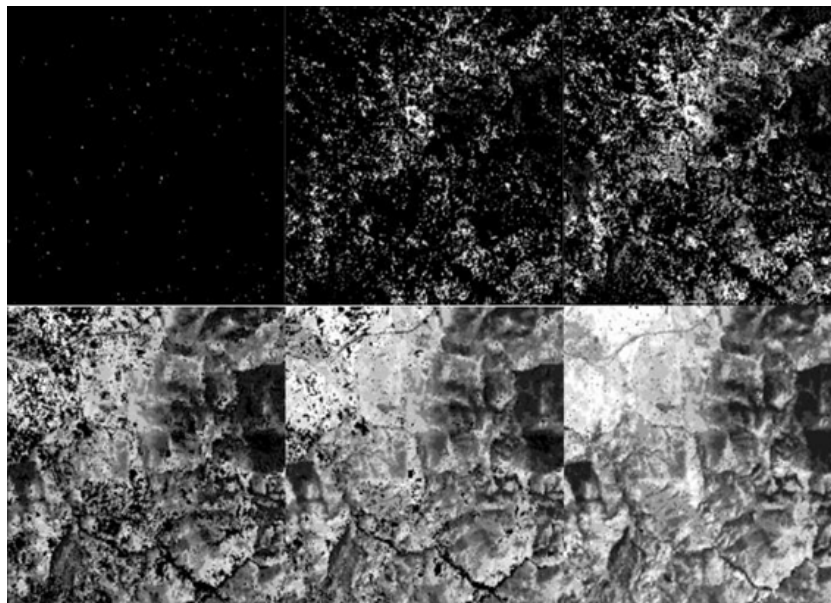


Figura 5.2: Clasificación jerárquica en 6 capas de fiabilidad.

La asignación de colores a cada clase se establece en la escala de grises, de modo que los píxeles negros son los que aún no han sido clasificados. Como se puede ver en la primera iteración, el número de píxeles clasificados es muy escaso. Estos píxeles son los que están más cerca del promedio de las clases espectrales definidas por el experto, y por lo tanto son las más fiables. Algunos de estos píxeles incluso pertenecen al conjunto de entrenamiento elegido por el experto. En las siguientes iteraciones el umbral de distancia al valor espectral medio de cada clase se va incrementando sucesivamente, de modo que los píxeles que se encuentran más alejados espectralmente de su clase correspondiente se van clasificando. En la última iteración, los píxeles inciertos y ruidosos son clasificados, ya que son usualmente los más problemáticos, basándose en la clasificación de los píxeles anteriores. Gracias a estos resultados, los expertos pueden detectar visualmente de una manera mucho más fiable los píxeles que son problemáticos en el proceso de clasificación (inciertos y ruidosos), información que resulta muy valiosa en el proceso de análisis de imágenes por satélite.

A continuación se muestran una serie de figuras que muestran con detalle el proceso de clasificación de la imagen de satélite mediante 100 iteraciones del algoritmo ACA. La Figura 5.3 muestra las 12 primeras iteraciones del proceso de clasificación, la Figura 5.4 muestra las iteraciones 13-24 y la Figura 5.5 las 75 iteraciones siguientes.

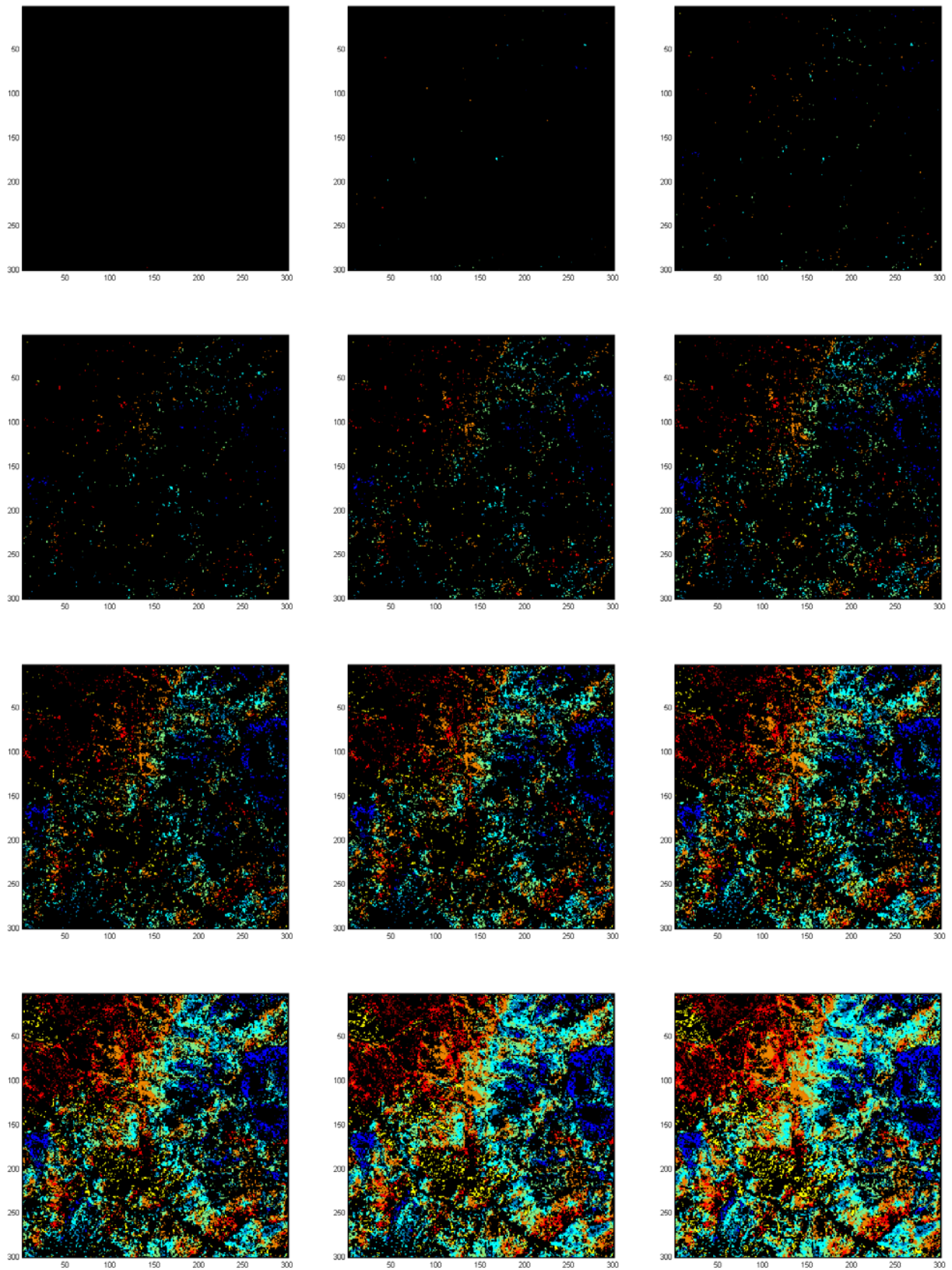


Figura 5.3: Clasificación obtenida en las iteraciones 1-12 del algoritmo ACA.



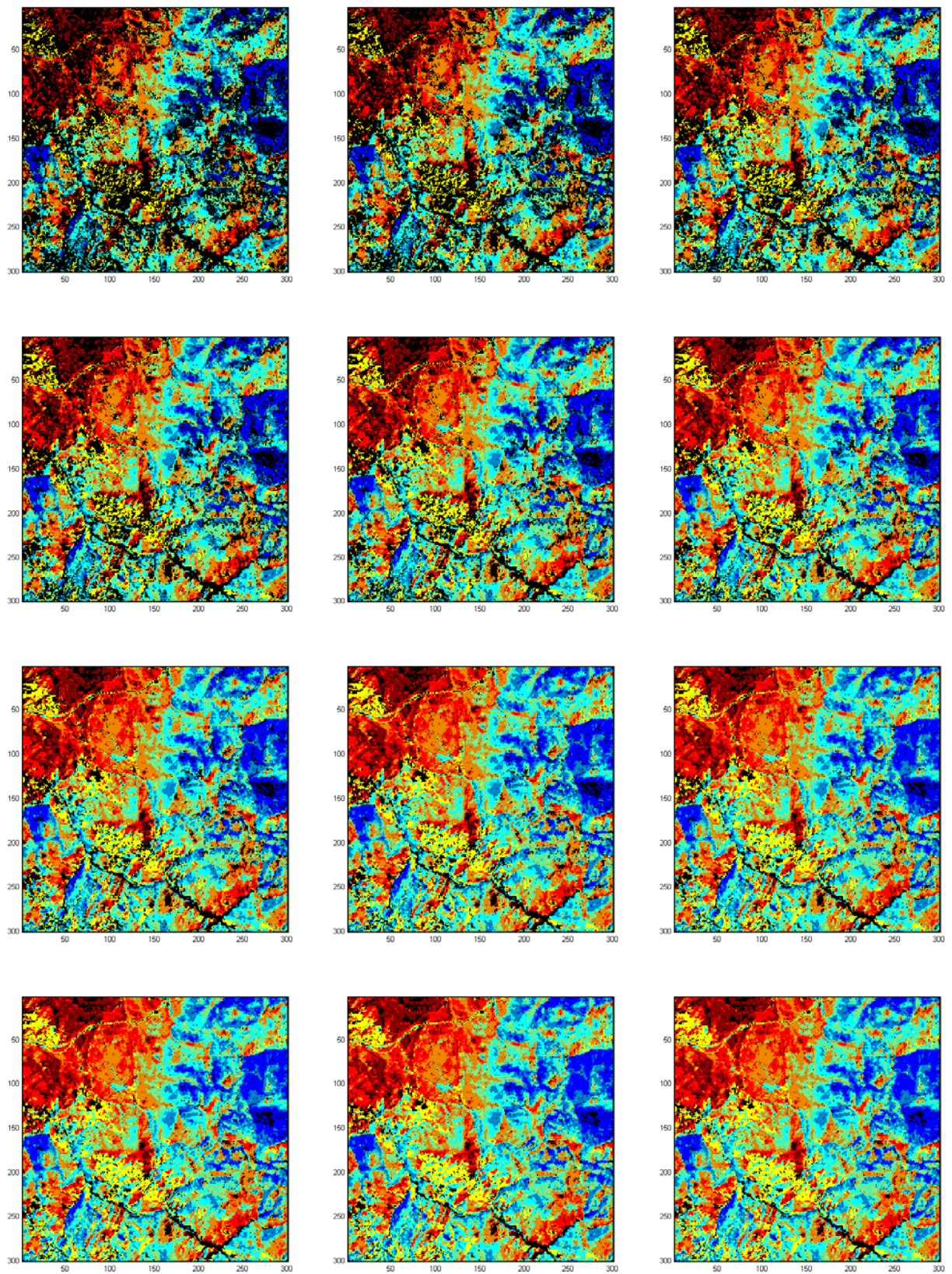


Figura 5.4: Clasificación obtenida en las iteraciones 13-24 del algoritmo ACA.



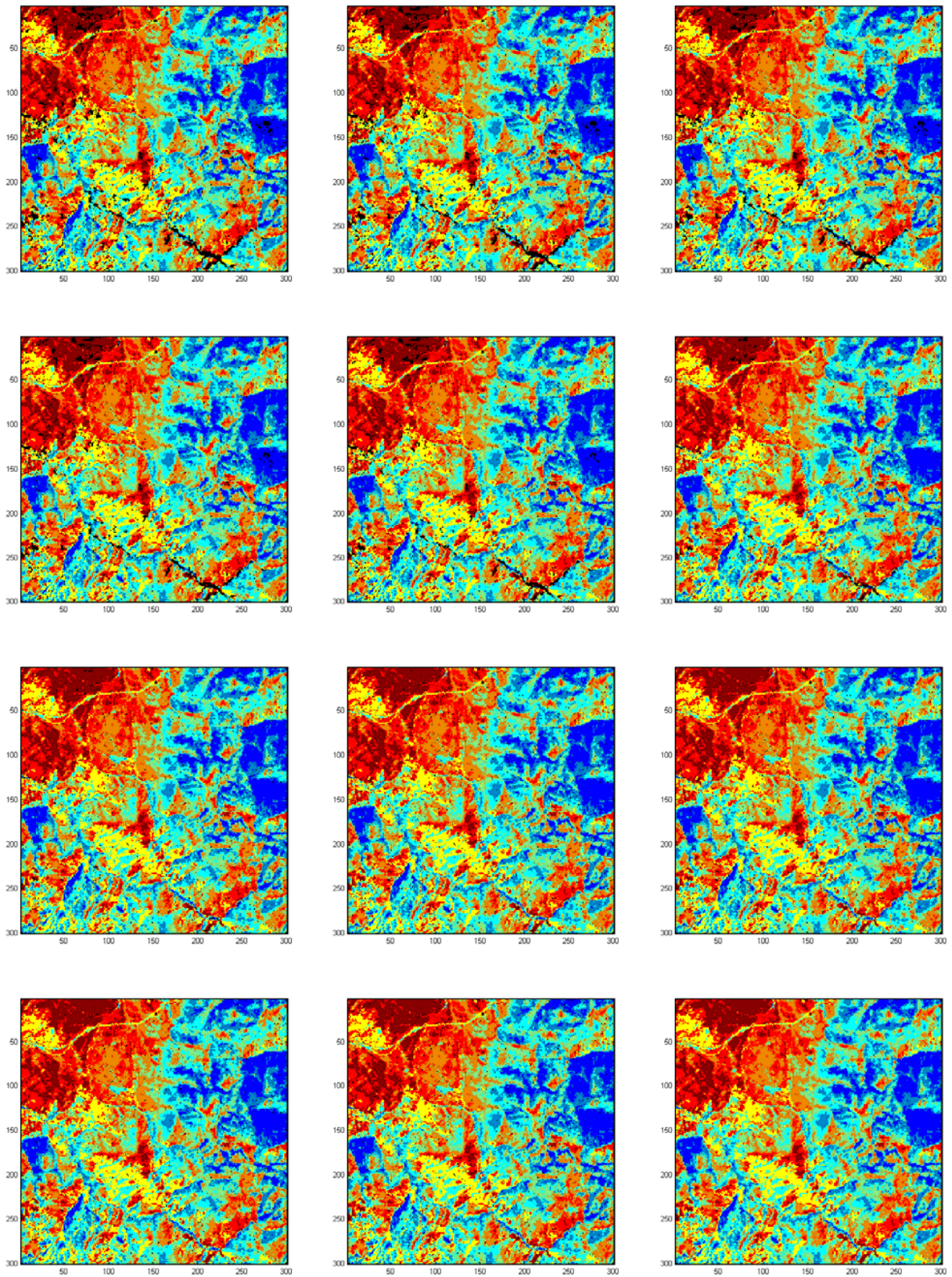


Figura 5.5: Clasificación obtenida en las iteraciones 26-30, 40, 50, 60, 70, 80, 90 y 100.

Como se puede observar, la mayoría de los píxeles son clasificados en las 15 primeras iteraciones y las siguientes iteraciones van refinando el proceso de clasificación para optimizar los resultados mediante técnicas contextuales.

El algoritmo ACA tarda un total de 13 segundos en realizar la clasificación de la imagen que hemos tomado como ejemplo. De esos 13 segundos, 10 son consumidos en la iteración 20. El resto, los otros 3 segundos, son consumidos en las 80 iteraciones restantes, que es donde se perfecciona el proceso de clasificación mediante técnicas contextuales para clasificar los píxeles inciertos y ruidosos. La Figura 5.6 muestra la distribución del tiempo necesario para clasificar la imagen de estudio mediante las 100 iteraciones anteriores del algoritmo ACA.

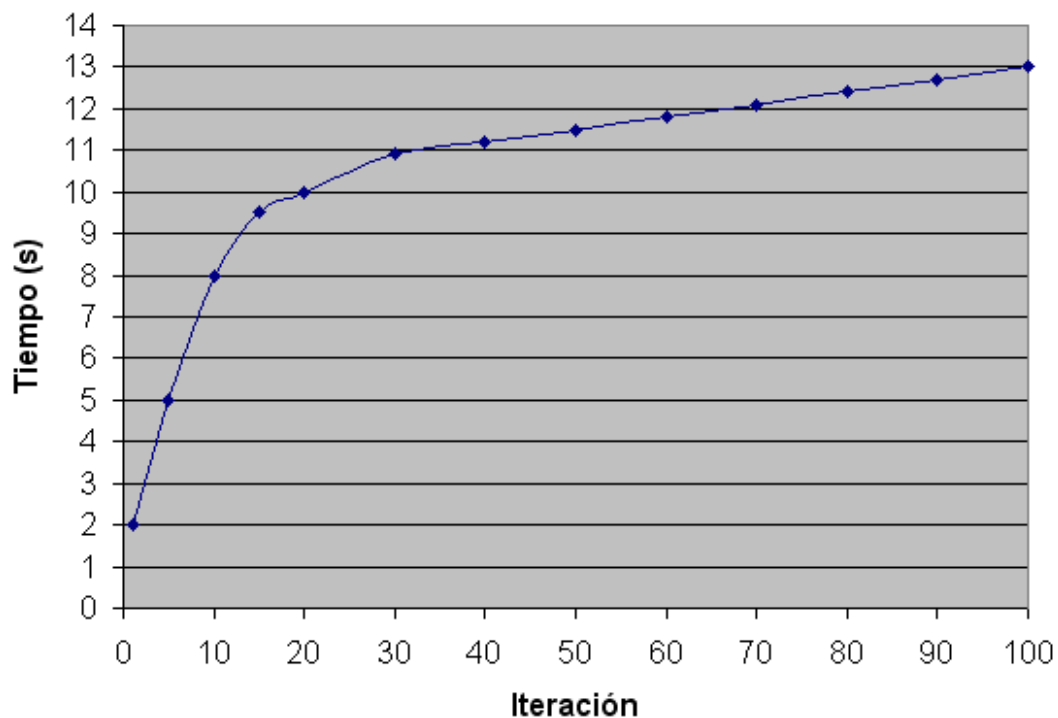


Figura 5.6: Gráfica: distribución del tiempo necesario para clasificar la imagen de estudio mediante 100 iteraciones del algoritmo ACA.

### 5.3. DETECCIÓN DE BORDES, PÍXELES INCIERTOS Y RUIDOSOS

Además, el algoritmo ACA también proporciona al experto una lista de los píxeles del borde espacial de cada clase representada en la imagen, así como los píxeles inciertos y ruidosos, con el fin de tener más información adicional relacionada con el proceso de clasificación para mejorar el análisis posterior de los resultados obtenidos.

Gracias a esto, se consigue extraer la máxima cantidad de datos posibles durante el proceso de clasificación, con el objetivo de que los expertos puedan disponer de mayor cantidad de información para el posterior proceso de análisis de la clasificación.

## 5.4. RESUMEN Y CONCLUSIONES DEL CAPÍTULO

Los resultados obtenidos en la clasificación de imágenes de satélite mediante autómatas celulares son muy satisfactorios desde varios puntos de vista.

En primer lugar, se consigue mejorar la calidad de la clasificación obtenida, un aspecto muy importante ya que en la actualidad no existe ningún algoritmo que ofrezca un 100 % de fiabilidad. Este objetivo se alcanza gracias a la información contextual que proporciona la vecindad del autómata celular durante el proceso de clasificación, mejorado por el hecho de usar vecinos clasificados en iteraciones anteriores a la actual, y por lo tanto píxeles más cercanos a las clases desde el punto de vista espectral. Esto provoca que los píxeles inciertos sean clasificados correctamente con mayor grado de acierto. Lo mismo ocurre con los píxeles ruidosos.

En segundo lugar, con el algoritmo ACA se obtiene una clasificación jerárquica basada en capas de fiabilidad, donde cada capa corresponde a una iteración del autómata celular. Como a lo largo de la ejecución del algoritmo ACA el radio espectral de pertenencia a las distintas clases va aumentando en cada iteración, las primeras iteraciones proporcionan capas de máxima fiabilidad, ya que en dichas capas se encuentran los píxeles más cercanos espectralmente a los centros de las clases (algunos incluso corresponden al conjunto de muestras). Conforme el autómata celular va ejecutando iteraciones, dicha distancia espectral aumenta, por lo tanto los píxeles que se clasifican tienen menos fiabilidad, por estar más alejados del centro de sus clases correspondientes. Por último, en las últimas iteraciones se clasifican aquellos píxeles de tipo incierto, en los que hace falta además información contextual para realizar un recuento de las clases de los píxeles vecinos (ya clasificados en iteraciones anteriores, y por lo tanto más fiables) y de este modo mejorar el resultado final.

En tercer lugar, el algoritmo también ofrece una detección de los bordes espaciales de las clases en la propia imagen de satélite, aspecto que puede ser de utilidad en la posterior interpretación y análisis de los resultados obtenidos, así como una lista de píxeles inciertos y ruidosos, para que los expertos puedan detectarlos con facilidad.

Todos estos resultados han sido publicados con gran aceptación en diversos congresos científicos internacionales, como se muestra en el siguiente capítulo.

---

## CAPÍTULO 6

# APORTACIONES Y TRABAJO FUTURO

---



# Capítulo 6

## APORTACIONES Y TRABAJO FUTURO

### Contenidos

---

<b>6.1.</b>	<b>PROYECTOS I+D . . . . .</b>	<b>101</b>
<b>6.2.</b>	<b>PUBLICACIONES . . . . .</b>	<b>102</b>
<b>6.3.</b>	<b>TRABAJO FUTURO . . . . .</b>	<b>104</b>
<b>6.4.</b>	<b>RESUMEN Y CONCLUSIONES DEL CAPÍTULO . . . . .</b>	<b>105</b>

---





**D**urante los años que he estado realizando trabajos de investigación, mi esfuerzo se ha centrado principalmente en las dos siguientes líneas:

- Autómatas celulares aplicados a la teledetección para realizar una clasificación contextual jerarquizada de imágenes de satélite, como miembro del Grupo de Investigación de Informática Aplicada (TIC-211), del departamento Lenguajes y Computación de la Universidad de Almería (Dr. Luis Iribarne).
- Computación gráfica 3D aplicada a proyectos I+D para realizar experimentos psicológicos con sujetos humanos, como miembro de la Unidad de Ingeniería en Computación Gráfica (UICG) del Grupo de Investigación de Sistemas de Información (TIC-192), del departamento Lenguajes y Computación de la Universidad de Almería (Dr. Jesús Almendros-Jiménez).

Este capítulo se centra en los objetivos que he conseguido durante mi periodo de investigación, y está dividido en los siguientes apartados principales:

- Proyectos I+D: en este apartado se detalla una lista de proyectos I+D en los que he participado a lo largo de estos años de investigación, relacionados con las dos líneas de investigación citadas anteriormente.
- Publicaciones: en este apartado se muestran los trabajos publicados durante mi fase de investigación, un total de 19 publicaciones (entre revistas, congresos, capítulos de libros y libros completos).
- Trabajo futuro: en este apartado se citan los futuros objetivos con los que pretendo perfilar mi trabajo de investigación para alcanzar la tesis doctoral.
- Resumen y conclusiones del capítulo: en este apartado se destacan los aspectos más importantes del trabajo conseguido durante mi período de investigación.

## 6.1. PROYECTOS I+D

En este apartado se detalla una lista de proyectos I+D en los que he participado a lo largo de estos años de investigación, relacionados con las dos líneas de investigación citadas anteriormente:

- *Desarrollo de un agente Web inteligente de información medioambiental* (TIC-6114) del grupo de investigación de Informática Aplicada (TIC-211) del departamento de Lenguajes y Computación de la universidad de Almería.
- *iSOLERES: una metodología para la recuperación y explotación de información medioambiental mediante interfaces de usuario evolutivas y cooperativas* (TIN2010-15588) del grupo de investigación de Informática Aplicada (TIC-211) del departamento de Lenguajes y Computación de la universidad de Almería.
- *SOLERES: un sistema de información espacio-temporal para la gestión medioambiental basado en redes neuronales, agentes y componentes software* (TIN2007-61497) del grupo de investigación de Informática Aplicada (TIC-211) del departamento de Lenguajes y Computación de la universidad de Almería.

- *EVEMEH 3: un simulador de realidad virtual 3D para la evaluación de memoria espacial en humanos en nuevos escenarios* en colaboración con el grupo de investigación de Psicofarmacología, Neurotoxicología y Neuropsicología (CTS-280) del departamento de Neurociencia y Ciencias de la Salud de la Universidad de Almería.
- *MII SCHOOL: comorbilidad psiquiátrica, acoso escolar y drogadicción en centros educativos: desarrollo de un soporte informático para su detección* en colaboración con el grupo de investigación de Análisis Clínico y Experimental de los Trastornos del Espectro Esquizofrénico (HUM-760) del departamento de Personalidad, Evaluación y Tratamiento Psicológico de la universidad de Almería.
- *EVEMEH 2: desarrollo de nuevos entornos virtuales 3D para la evaluación de memoria espacial en humanos* en colaboración con el grupo de investigación de Psicofarmacología, Neurotoxicología y Neuropsicología (CTS-280) del departamento de Neurociencia y Ciencias de la Salud de la Universidad de Almería.
- *DRAZOR: estudio y diseño de sistemas para animación, comportamiento y robotizado de juegos inmersivos 3D* en colaboración con la empresa Imagital de Almería.
- *EVEMEH 1: estudio de un entorno virtual 3D para la evaluación de memoria espacial en humanos* en colaboración con el grupo de investigación de Psicofarmacología, Neurotoxicología y Neuropsicología (CTS-280) del departamento de Neurociencia y Ciencias de la Salud de la Universidad de Almería.

En los tres primeros proyectos I+D (TIC-6114, TIN2010-15588 y TIN2007-61497) he investigado aspectos relacionados con autómatas celulares aplicados al ámbito de la teledetección para clasificación de imágenes de satélite. En los cinco últimos proyectos he desarrollado aplicaciones relacionadas con la computación gráfica 3D para realizar estudios de psicología con sujetos de experimentación humanos. En el siguiente apartado se exponen las publicaciones obtenidas en ambas líneas de investigación: teledetección y computación gráfica 3D.

## 6.2. PUBLICACIONES

En este apartado se muestran los trabajos publicados durante mi fase de investigación, un total de 19 publicaciones (entre revistas, congresos, capítulos de libros y libros completos). A continuación se enumeran las publicaciones relacionadas con el trabajo que he llevado a cabo en el ámbito de la teledetección durante mi período de investigación, y que corresponden al trabajo expuesto en el presente documento:

1. Cruz, M., **Espínola, M.**, Iribarne, L., Ayala, R., Peralta, M., Torres, J.A.: How can neural networks to speed up ecological regionalization friendly? Replacement of field studies by satellite data using RBFs. International Conference on Neural Computation (ICNC 2010), october 24-26, Valencia (Spain), (2010)
2. **Espínola, M.**, Ayala, R., Leguizamón, S., Iribarne, L., Menenti, M.: Cellular automata applied in remote sensing to implement contextual pseudo-fuzzy classification. The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010), september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 6350, pp. 312–321 (2010).

3. Leguizamón, S., **Espínola, M.**, Ayala, R., Iribarne, L., Menenti, M.: Characterization of texture in images by using a cellular automata approach. Third World Summit on the Knowledge Society (WSKS 2010), september 22-24, Corfu (Greece), in M.D. Lytras et al. eds, *The Open Knowledge Society: Organizational, Business, and Technological Aspects of the Knowledge Society*, Springer Communications in Computer and Information Science 112, part II, pp. 522–533 (2010).
4. Cruz, M., **Espínola, M.**, Iribarne, L., Ayala, R., Peralta, M., Torres, J.A.: Ecological sectorization process improvement through neural networks: synthesis of vegetation data from satellite images using RBFs. 9th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2010), august 18-20, Yamagata (Japan), pp. 113–116 (2010).
5. **Espínola, M.**, Ayala, R., Leguizamón, S., Menenti, M.: Classification of satellite images using the cellular automata approach. First World Summit on the Knowledge Society (WSKS 2008), september 24-28, Athens (Greece), in M.D. Lytras et al. eds, *The Open Knowledge Society: A Computer Science and Information System Manifesto*, Springer Communications in Computer and Information Science 19, pp. 521–526 (2008).

Como se comento en el apartado anterior, durante estos años también he participado en diversos proyectos I+D relacionados con distintos ámbitos relacionados con la computación gráfica 3D aplicada a la psicología humana, y además he participado en publicación de varios libros didácticos:

1. Carmona, J.A., **Espínola, M.**, Cangas, A.J., Iribarne, L.: Applying virtual reality (VR) to the detection and treatment of clinical problems in educational settings. In J. Zhao (Ed.). *Technology Enhanced Learning for People with Disabilities: Approaches and Applications*. Information Science Publishing, pp. 194–202 (2011).
2. Carmona, J.A., **Espínola, M.**, Cangas, A.J., Iribarne, L.: Detecting drug use in adolescents using 3D simulation program. *Psychology, society and education*, pp. 61–71, (2010).
3. Carmona, J.A., **Espínola, M.**, Cangas, A.J., Iribarne, L.: Mii school: new 3D technologies applied in education to detect drug abuses and bullying in adolescents. 1st International Conference on Technology Enhanced Learning, Quality of Teaching and Educational Reform (TECH-EDUCATION 2010), Atenas (Grecia), CCIS 73, pp. 65–72, (2010).
4. Carmona, J.A., **Espínola, M.**, Cangas, A.J., Iribarne, L.: Mii School: una aplicación clínica de la simulación 3D para la evaluación de conductas problemáticas en jóvenes e implicaciones para el tratamiento. Congreso Internacional sobre Avances en Tratamientos Psicológicos, Granada (España), abril 14-17 (2010).
5. **Espínola, M.**: Making of Mii School. *Blender Art Magazine*, Issue 26: Blender & Gaming, pp. 30-35, (2010).
6. Carmona, J.A., **Espínola, M.**, Cangas, A.J., Iribarne, L.: Mii-School: a 3D videogame for the early detection of abuse of substances, bullying and mental disorders in adolescents. *European Journal of Education and Psychology*, vol 3 (ISSN 1888-8992), n° 1, (2010).

7. Cangas, A.J., Carmona, J.A., **Espínola, M.**: Un programa de ordenador para detectar conductas de riesgo en adolescentes. *Andalucía Investiga*, 1 diciembre, (2008).
8. Cánovas, R., **Espínola, M.**, Iribarne, L., Cimadevilla, J.M.: A new virtual task to evaluate human place learning. *Behavioural Brain Research Journal* 190, pp. 112-118, doi:10.1016/j.bbr.2008.02.024, (2008).
9. Gómez, J., **Espínola, M.**: Manual de redes inalámbricas. ISBN: 84-96270-97-8, 200 páginas, (2007).
10. Iribarne, L., **Espínola, M.**, Berenguel, J.L., Márquez, A.: Programación de animaciones en 3D: un motor gráfico en OpenGL. ISBN: 84-96270-89-3, 200 páginas, (2007).
11. Cánovas, R., Iribarne, L., **Espínola, M.**, León, I., Roldán, D., Cimadevilla, J.M.: A new virtual task to evaluate human place learning. *International Neuropsychological Society, Federation of Spanish Societies of Neuropsychology, Spanish Neuropsychological Society, Spanish Psychiatry Society Joint Mid-Year Meeting, July 4-7, Bilbao (Spain)*, pp. 43-43, (2007).
12. **Espínola, M.**, Parra, G., Iribarne, L., Torres, J.A.: Designing virtual tours of 3D-adaptative greenhouses. *Ibero-American Symposium in Computer Graphics (SIACG 2006)*, July 5-7, Santiago de Compostela (Spain), pp. 28-31, (2006).
13. Iribarne, L., **Espínola, M.**, Parra, G.: Iniciación al diseño y programación de videojuegos 3D. ISBN: 84-96270-69-6, 150 páginas, (2006).
14. Iribarne, L., **Espínola, M.**: Análisis y diseño del software. Teoría y práctica. ISBN: 84-96270-53-X, 442 páginas, (2005).

### 6.3. TRABAJO FUTURO

En este apartado se citan los futuros objetivos con los que pretendo perfilar mi trabajo de investigación para alcanzar la tesis doctoral:

- Desarrollar nuevas versiones del algoritmo de ACA con una configuración de estados y reglas del autómata celular distintas para personalizar aún más el proceso de clasificación. Desarrollo de un algoritmo clasificador ACA difuso.
- Añadir un nuevo nivel de clasificación al algoritmo ACA: la clasificación textural (basada en texturas). De este modo, tendríamos dos niveles distintos de clasificación: nivel de píxel (espectral y contextual) y nivel regional (textural).
- Utilizar agentes software para reducir el coste computacional del algoritmo ACA, recorriendo y procesando varias regiones de la imagen en paralelo a lo largo de las iteraciones en las que se divide el proceso de clasificación.
- Desarrollar un plugin con una interfaz gráfica y un editor de reglas-estados del algoritmo ACA para la herramienta Erdas Imagine que permita una clasificación personalizada basada en autómatas celulares.

## 6.4. RESUMEN Y CONCLUSIONES DEL CAPÍTULO

Durante el período de investigación he realizado la asignatura “Modelado de sistemas de información” de 12 créditos tutelada por los profesores Luis F. Iribarne Martínez y Jesús Manuel Almendros Jiménez. El resultado de todos los trabajos de investigación realizados, y sobre todo el relacionado con el tema de mi tesis doctoral presentado en este documento, ha tenido buena acogida por el comité científico internacional, ya que ha sido publicado en varias revistas y congresos internacionales.

Aunque todavía queda bastante trabajo por realizar para alcanzar todos los objetivos que me he propuesto para completar la investigación completa de mi tesis doctoral, considero que los resultados obtenidos hasta el momento son lo suficientemente satisfactorios como para continuar con esta línea de investigación.



---

PROPUESTA DE ADSCRIPCIÓN  
A UN AREA DE CONOCIMIENTO

---





## PROPUESTA DE ADSCRIPCIÓN A UN ÁREA DE CONOCIMIENTO

Propongo formalmente que el trabajo de investigación que he presentado en esta memoria sea adscrito al Área de conocimiento de Lenguajes y Sistemas Informáticos (LSI), perteneciente al Departamento de Lenguajes y Computación de la Universidad de Almería, dentro de las líneas de Investigación de Modelado de Sistemas de Información e Ingeniería del Software.

Moisés Espínola Pérez  
Grupo de investigación de Informática Aplicada  
Departamento de Lenguajes y Computación  
Universidad de Almería  
ALMERÍA, 2011



---

## BIBLIOGRAFÍA

---



# Bibliografía

- [1] Aponte, A., Moreno, J.A.: Cellular automata and its application to the modeling of vehicular traffic in the city of Caracas. 7th International Conference on Cellular Automata for Research and Industry (ACRI 2006), september 20-23, Perpignan (France), in S. El Yacoubi et al. eds, Springer Lecture Notes in Computer Science 4173, pp. 502–511, (2006).
- [2] Asamblea General de las Naciones Unidas: Principios relativos a la teleobservación de la Tierra desde el espacio. Resolución 41/65 de 1986, principio I, pag. 120, (1986).
- [3] Avolio, M.V., Errera, A., Lupiano, V., Mazzanti, P., Di Gregorio, S.: Development and calibration of a preliminary cellular automata model for snow avalanches. The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010), september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 6350, pp. 83–94, (2010).
- [4] Ayala, R., Menenti, M., Girolana D.: Evaluation methodology for classification process of digital images Igarss '02. IEEE Int. Geoscience and Remote Sensing Symposium and the 24th Canadian Symposium on Remote Sensing. Toronto, Canada, pp. 3363–3365, (2002).
- [5] Ayala, R., Becerra, A., Flores, I.M, Bienvenido, J.F, Díaz, J.R.: Evaluation of greenhouse covered extensions and required resources with satellite images and GIS. Almería's case. Second European Conference of the European Federation for Information Technology in Agriculture, Food and the Environment. Bonn, Germany, pp. 27–30, (1999).
- [6] Balzter, H., Braun, P., Kühler, W.: Cellular automata models for vegetation dynamics. *Ecological Modelling*, 107, pp. 113–125, (1998).
- [7] Bandini, S., Bonomi, A., Vizzari, G.: A cellular automata based modular illumination system. The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010), september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 6350, pp. 334–344, (2010).
- [8] Barberousse, A., Franceschelli, S., Imbert, C.: Cellular automata, modeling, and computation. *Models and Simulations 2* (Tilburg, NL), (2007).
- [9] Barret, E.C., Curtis L.F.: Introduction to environmental remote sensing. Cheltenham, Stanley Thornes Publishers Ltd, (1999).
- [10] Blečić, I., Cecchini, A., Rizzi, P., Trunfio, G.A.: Playing with automata. An innovative perspective for gaming simulation. 5th International Conference on Cellular Automata for Research and Industry (ACRI 2002), october 9-11, Geneva (Switzerland), in S.

- Bandini et al. eds, Springer Lecture Notes in Computer Science 2493, pp. 337–348, (2002).
- [11] Castro, A., Gómez, N.: Self-organizing map and cellular automata combined technique for advanced mesh generation in urban and architectural design. *International Journal of Information Technologies and Knowledge*, vol 2, pp. 354–360, (2008).
- [12] Cheng, J., Masser, I.: Cellular automata based temporal process understanding of urban growth. 5th International Conference on Cellular Automata for Research and Industry (ACRI 2002), october 9-11, Geneva (Switzerland), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 2493, pp. 325–336, (2002).
- [13] Ching Fu, S., Milne, G.: A flexible automata model for disease simulation. 6th International Conference on Cellular Automata for Research and Industry (ACRI 2004), october 25-27, Amsterdam (The Netherlands), in P.M.A. Sloot et al. eds, Springer Lecture Notes in Computer Science 3305, pp. 642–649, (2004).
- [14] Chopard, B., Lagrava, D.: A cellular automata model for species competition and evolution. 7th International Conference on Cellular Automata for Research and Industry (ACRI 2006), september 20-23, Perpignan (France), in S. El Yacoubi et al. eds, Springer Lecture Notes in Computer Science 4173, pp. 277–286, (2006).
- [15] Chuvieco, E., Huete, A.: *Fundamentals of satellite remote sensing*. CRC Press, Boca Raton (USA), (2010).
- [16] Cruz, M., Espínola, M., Iribarne, L., Ayala, R., Peralta, M., Torres, J.A.: How can neural networks to speed up ecological regionalization friendly? Replacement of field studies by satellite data using RBFs. *International Conference on Neural Computation (ICNC 2010)*, october 24-26, Valencia (Spain), (2010).
- [17] Cruz, M., Espínola, M., Iribarne, L., Ayala, R., Peralta, M., Torres, J.A.: Ecological sectorization process improvement through neural networks: synthesis of vegetation data from satellite images using RBFs. *9th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2010)*, august 18-20, Yamagata (Japan), pp. 113–116, (2010).
- [18] Das, S., Chowdhury, D.R.: Generating cryptographically suitable non linear maximum length cellular automata. *The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010)*, september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 6350, pp. 241–250, (2010).
- [19] Dzwiniel, W.: A cellular automata model of population infected by periodic plague. 6th International Conference on Cellular Automata for Research and Industry (ACRI 2004), october 25-27, Amsterdam (The Netherlands), in P.M.A. Sloot et al. eds, Springer Lecture Notes in Computer Science 3305, pp. 464–473, (2004).
- [20] Espínola, M., Ayala, R., Leguizamón, S., Iribarne, L., Menenti, M.: Cellular automata applied in remote sensing to implement contextual pseudo-fuzzy classification. *The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010)*, september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 6350, pp. 312–321, (2010).

- [21] Espínola, M., Ayala, R., Leguizamón, S., Menenti, M.: Classification of satellite images using the cellular automata approach. First World Summit on the Knowledge Society (WSKS 2008), september 24-28, Athens (Greece), in M.D. Lytras et al. eds, *The Open Knowledge Society: A Computer Science and Information System Manifesto*, Springer Communications in Computer and Information Science 19, pp. 521–526, (2008).
- [22] Fúster-Sabater, A., Caballero-Gil, P.: Chaotic cellular automata with cryptographic application. The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010), september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, *Springer Lecture Notes in Computer Science* 6350, pp. 251–260, (2010).
- [23] Gardner, M.: The fantastic combination of john conways new solitaire game “Life”. *Scientific American* 223(4), pp. 120–123, (1970).
- [24] Ghaemi, M., Naderi, O., Zabihinpour, Z.: A novel method for simulating cancer growth. The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010), september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, *Springer Lecture Notes in Computer Science* 6350, pp. 142–148, (2010).
- [25] Guzman, O.F., Gómez, I.D.: *Fundamentos físicos de teledetección*. Instituto Geografico Agustin Codazzi, IGAC, (2007).
- [26] Haralick, R.M.: Statistical and structural approaches to texture. *Proceedings of the IEEE* 67, pp. 786–804, (1979).
- [27] Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. *IEEE Trans. On System, Man, and Cybernetics*, SMC-3(6), pp. 610–621, (1973).
- [28] He, D.C., Wang, L.: Texture unit, texture spectrum, and texture analysis. *IEEE Trans. On Geoscience and Remote Sensing* 28(4), (1990).
- [29] Karafyllidis, I., Thanailakis, A.: A model for predicting forest fire spreading using cellular automata. *Ecological Modeling* 99, pp. 87–97, (1997).
- [30] Kari, J.: Theory of cellular automata: a survey. *Theoretical Computer Science* 334, pp. 3–33, (2005).
- [31] Leguizamón, S., Espínola, M., Ayala, R., Iribarne, L., Menenti, M.: Characterization of texture in images by using a cellular automata approach. Third World Summit on the Knowledge Society (WSKS 2010), september 22-24, Corfu (Greece), in M.D. Lytras et al. eds, *The Open Knowledge Society: Organizational, Business, and Technological Aspects of the Knowledge Society*, Springer Communications in Computer and Information Science 112, part II, pp. 522–533, (2010).
- [32] Leguizamón S.: Modeling land features dynamics by using cellular automata techniques. *Proceedings of the ISPR Technical Comision 7 Mid-Term Symposium “From pixels to Processess”*. Enschede, The Netherlands, pp. 497–501, (2006).
- [33] Leguizamón S.: Simulation of snow-cover dynamics using the cellular automata approach. *Proceedings of the 8th International Symposium on High Mountain Remote Sensing Cartography*. La Paz, Bolivia, pp. 87–91, (2005).
- [34] Leguizamón, S.: Characterization of texture in remotely sensed images by using the wavelet transform. *Proceedings of the IV International Symposium on HMRS Cartography*. University of Karlstad, Karlstad, (1996).

- [35] Leguizamón, S.: Description of terrain textures by fractal and Markov random fields techniques. Proceedings of the 2nd. International Symposium on HMRS Cartography, Chinese Academy of Sciences, Astronautic Publ. House, P.R. of China, Beijing, (1993).
- [36] Lobitz, B., Beck, L., Huq, A., Woods, B., Fuchs, G., Faruque, A., Colwell, R.: Climate and infectious disease: use of remote sensing for detection of vibrio cholerae by indirect measurement. Proceedings of the National Academic of Sciences of the USA, 97(4), pp. 1438–43, (2000).
- [37] Maji, P., Shaw, C., Ganguly, N., Sikdar, B., Chaudhuri, P.: Theory and application of cellular automata for pattern classification. *Fundamenta Informaticae*, 58(3-4), pp. 321–354, (2003).
- [38] Marcou, O., El Yacoubi, S., Chopard, B.: A bi-fluid lattice boltzmann model for water flow in an irrigation channel. 7th International Conference on Cellular Automata for Research and Industry (ACRI 2006), september 20-23, Perpignan (France), in S. El Yacoubi et al. eds, Springer Lecture Notes in Computer Science 4173, pp. 373–382, (2006).
- [39] Mather, P., Tso, B.: Classification methods for remotely sensed data. CRC Press, 2 edition, (2009).
- [40] Messina, J., Walsh, S.: Simulating land use and land cover dynamics in the ecuadorian Amazon through cellular automata approaches and an integrated GIS. Open Meeting of the Human Dimensions of Global Environmental Change Research Community in Rio de Janeiro, Brazil, pp. 6–8, (2001).
- [41] Mojaradi, B., Lucas, C., Varshosaz, M.: Using learning cellular automata for post classification satellite imagery. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 35(4), pp. 991–995, (2004).
- [42] Moreno, N., Quintero, R., Ablan, M., Barros, R., Dávila, J., Ramírez, H., Tonella, G., Acevedo, M.: Biocomplexity of deforestation in the Caparo tropical forest reserve in Venezuela: an integrated multi-agent and cellular automata model. *Environmental Modelling and Software*, 22, pp. 664–673, (2007).
- [43] Muzy, A., Innocenti, E., Aiello, A., Santucci, J.F., Santonio, P.A., Hill, D.: Modelling and simulation of ecological propagation processes: application to fire spread. *Environmental Modelling and Software*, 20, pp. 827–842, (2005).
- [44] Muzy, A., Innocenti, E., Aiello, A., Santucci, J.F., Santonio, P.A., Hill, D.: Dynamic structure cellular automata in a fire spreading application. First International Conference on Informatics in Control, Automation and Robotics. Setubal, Portugal, pp. 143–151, (2004).
- [45] Popovici, A., Popovici, D.: Cellular automata in image processing. Proceedings of the 15th International Symposium on the Mathematical Theory of Networks and Systems, (2002).
- [46] Quartieri, J., Mastorakis, N.E., Iannone, G., Guarnaccia, C.: Cellular automata application to traffic noise control. Proceedings of the 12th WSEAS International Conference on Automatic Control, Modeling and Simulation (ACMOS10), Catania (Italy), pp. 29–31, (2010).



- [47] Rees, W.G.: Physical principles of remote sensing. Cambridge University Press, 2nd Edition, (2001).
- [48] Schonfisch, B., Kinder, M.: A fish migration model. 5th International Conference on Cellular Automata for Research and Industry (ACRI 2002), october 9-11, Geneva (Switzerland), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 2493, pp. 210–219, (2002).
- [49] Schowengerdt, R.A.: Techniques for image processing and classification in remote sensing. Academic Press, (1985).
- [50] Sloot, P., Chen, F., Boucher, C.: Cellular automata model of drug therapy for HIV infection. 5th International Conference on Cellular Automata for Research and Industry (ACRI 2002), october 9-11, Geneva (Switzerland), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 2493, pp. 282–293, (2002).
- [51] Spataro, W., D. Ambrosio, D., Rongo, R., Trunfio, G.A.: An evolutionary approach for modelling lava flows through cellular automata. 6th International Conference on Cellular Automata for Research and Industry (ACRI 2004), october 25-27, Amsterdam (The Netherlands), in P.M.A. Sloot et al. eds, Springer Lecture Notes in Computer Science 3305, pp. 725–734, (2004).
- [52] Toffoli, T., Margolus, N.: Invertible Cellular Automata: A Review. *Physica D* 45, pp. 229–253, (1990).
- [53] Topa, P.: Towards a two-scale cellular automata model of tumour-induced angiogenesis. 7th International Conference on Cellular Automata for Research and Industry (ACRI 2006), september 20-23, Perpignan (France), in S. El Yacoubi et al. eds, Springer Lecture Notes in Computer Science 4173, pp. 337–346, (2006).
- [54] Tsiftsis, A., Sirakoulis, G., Lygouras, J.: Design of a cellular automaton model for railway traffic flow with GPS module. The Ninth International Conference on Cellular Automata for Research and Industry (ACRI 2010), september 21-24, Ascoli Piceno (Italy), in S. Bandini et al. eds, Springer Lecture Notes in Computer Science 6350, pp. 373–384, (2010).
- [55] Umarani, C., Ganesan, L., Radhakrishnan, S.: A combined statistical and structural approach for texture representation. *Asia J. Inform. Technol.* 5, pp. 1434–1440, (2006).
- [56] Wang, J., Kropff, M., Lammert, B., Christensen, S., Hansen, P.: Using CA model to obtain insight into mechanism of plant population spread in a controllable system: annual weeds as an example. *Ecological Modelling*, 166, pp. 277–286, (2003).
- [57] Wolfram, S.: A new kind of science. Wolfram Media, Inc., Champaign, Illinois, (2002).
- [58] Wolfram, S.: Universality and complexity in cellular automata. *Physica D*, 10, pp. 1–35, (1984).
- [59] Wolfram, S.: Cellular automata. *Los Alamos Science*, 9, pp. 2–21, (1983).
- [60] Wolfram, S.: Cellular automata as simple self-organizing systems. Caltech preprint CALT, pp. 68–938, (1982).

- [61] Xia, Z., Jiang, Y.: A novel artificial life ecosystem environment model. 6th International Conference on Cellular Automata for Research and Industry (ACRI 2004), october 25-27, Amsterdam (The Netherlands), in P.M.A. Sloot et al. eds, Springer Lecture Notes in Computer Science 3305, pp. 650–659, (2004).

Este documento ha sido generado  
con la versión LaTeX2e de L<sup>A</sup>T<sub>E</sub>X

Clasificación de imágenes de satélite mediante Autómatas Celulares

A. Moisés Espínola Pérez  
Grupo de investigación de Informática Aplicada  
Departamento de Lenguajes y Computación  
Universidad de Almería  
Almería, 1 de Marzo de 2011  
<http://www.ual.es/personal/moises.espinola/>



