

AUTÓMATAS CELULARES MULTIESTADO APLICADOS A LA CLASIFICACIÓN CONTEXTUAL ITERATIVA DE IMÁGENES DE SATÉLITE

**A. Moisés Espínola Pérez
- PROYECTO SOLERES -**

24 de febrero de 2011

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

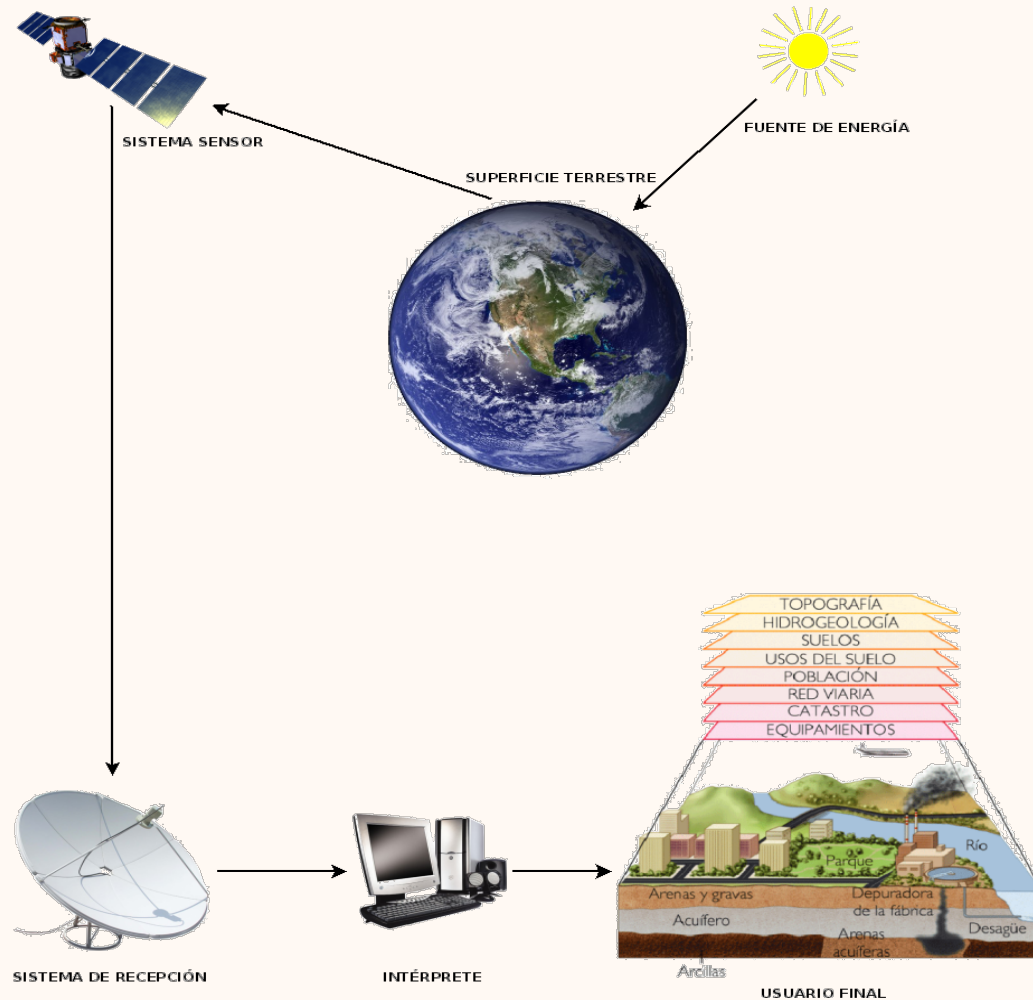
- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

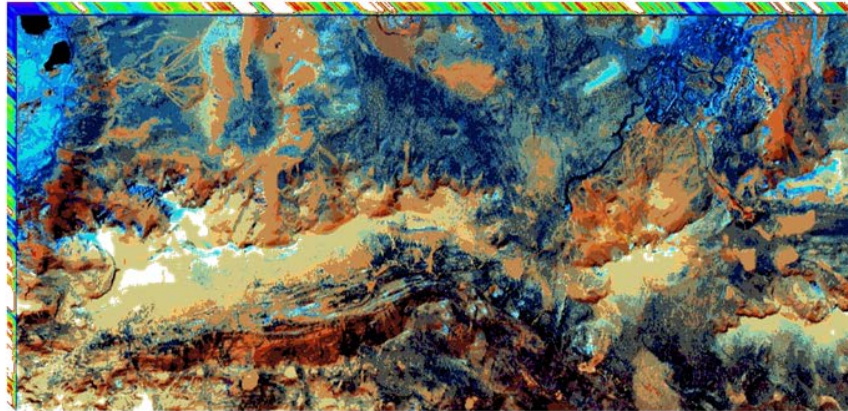
- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

TELEDETECCIÓN

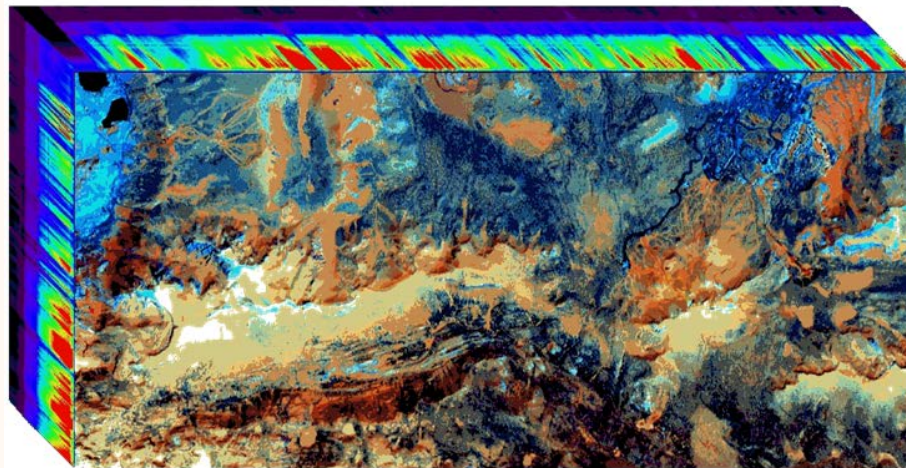
ELEMENTOS BÁSICOS DE LA TELEDETECCIÓN



IMÁGENES DE SATÉLITE MULTIESPECTRALES E HIPERESPECTRALES

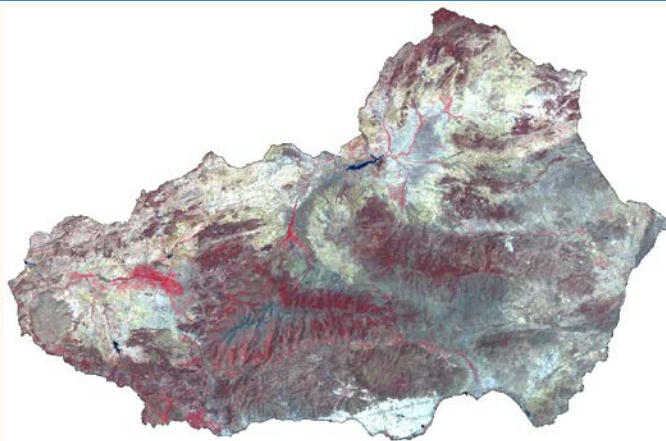


(a)



(b)

BANDAS DE LAS IMÁGENES DE SATÉLITE

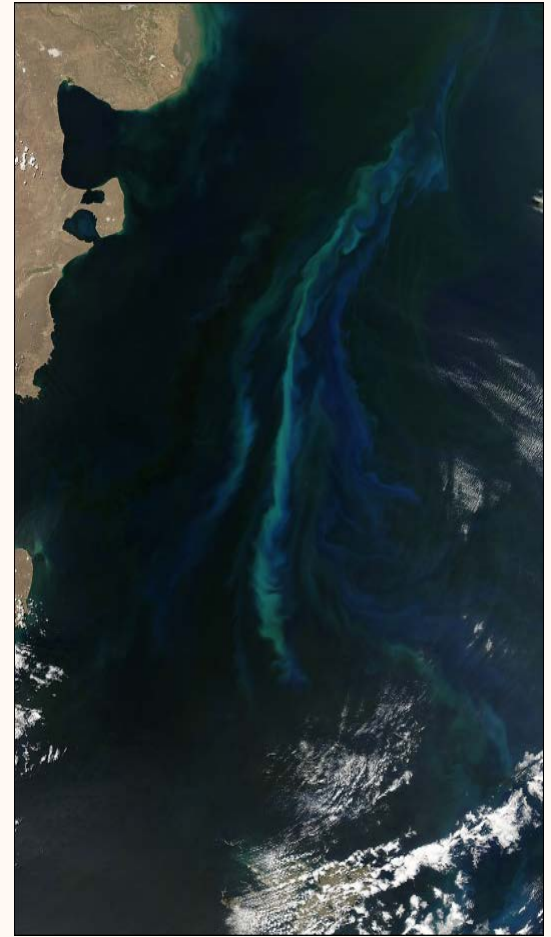


(a)



(b)

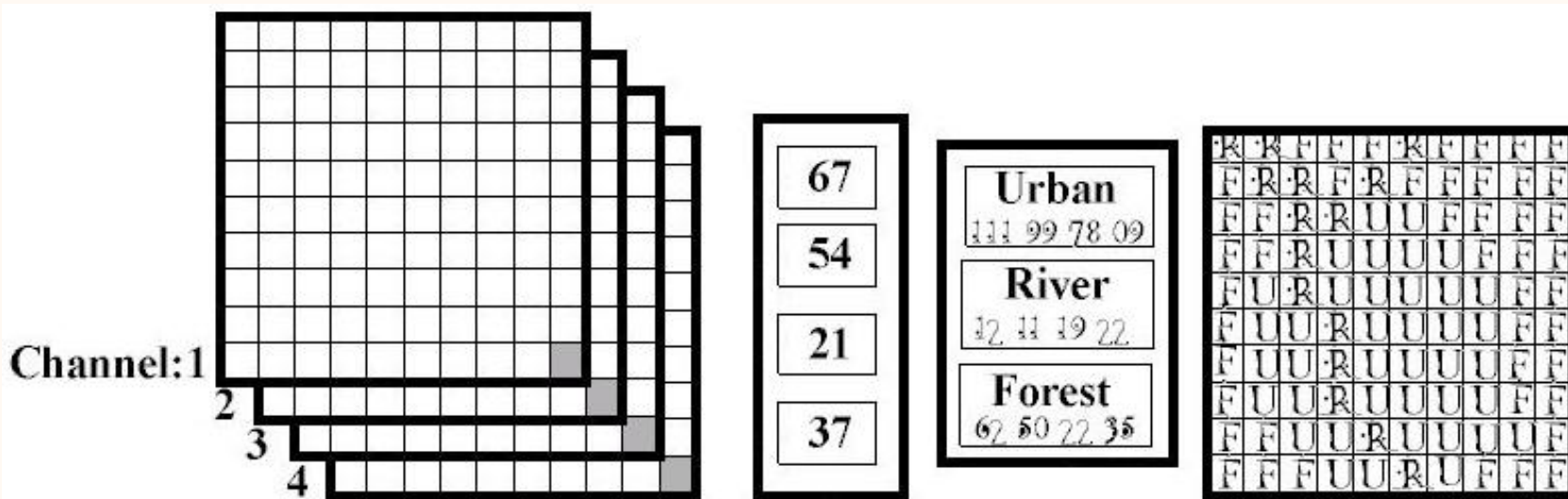
APLICACIONES DE LA TELEDETECCIÓN



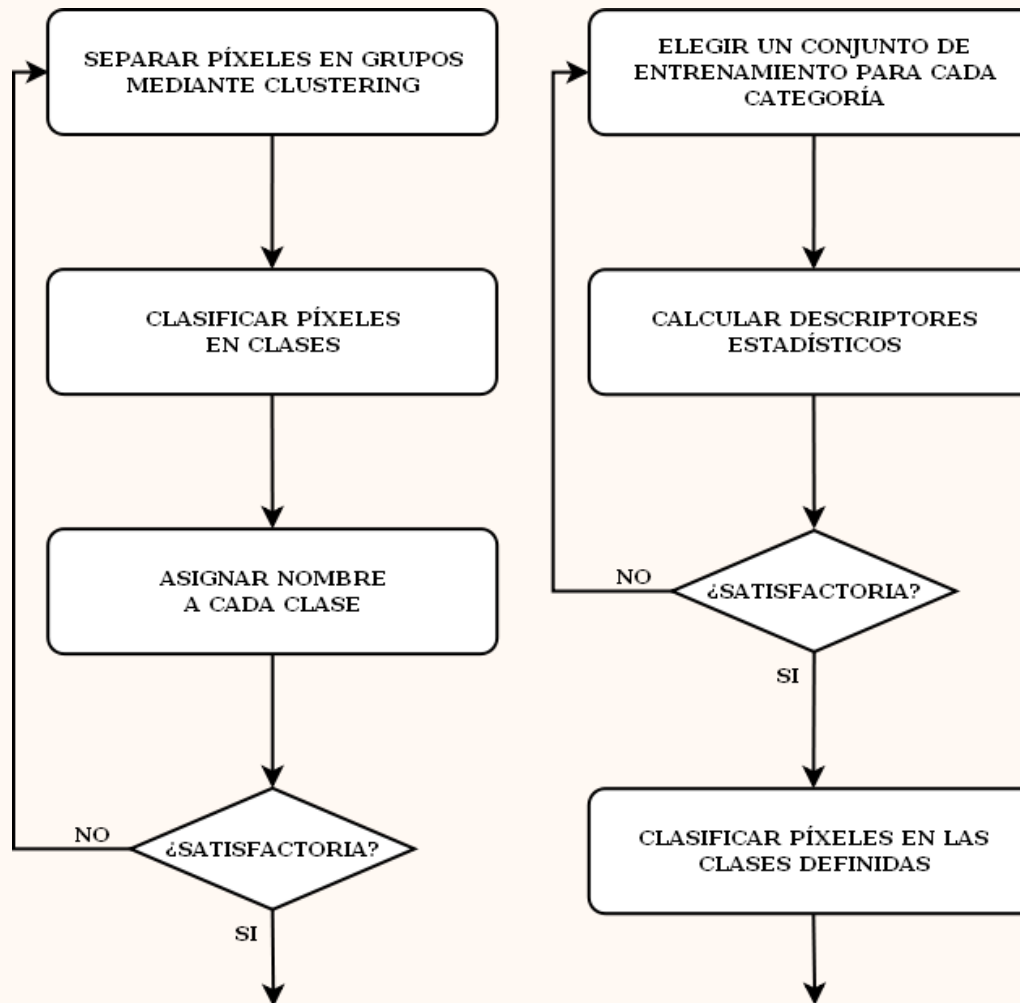
PROYECTO SOLERES, equipo de imágenes de satélite

A. Moisés Espínola Pérez

ALGORITMOS DE CLASIFICACIÓN DE IMÁGENES DE SATÉLITE



ALGORITMOS DE CLASIFICACIÓN SUPERVISADOS Y NO SUPERVISADOS



ALGORITMOS DE CLASIFICACIÓN MÁS COMUNES

**Clasificación
no supervisada**

- **K-medias**

- **Isodata**

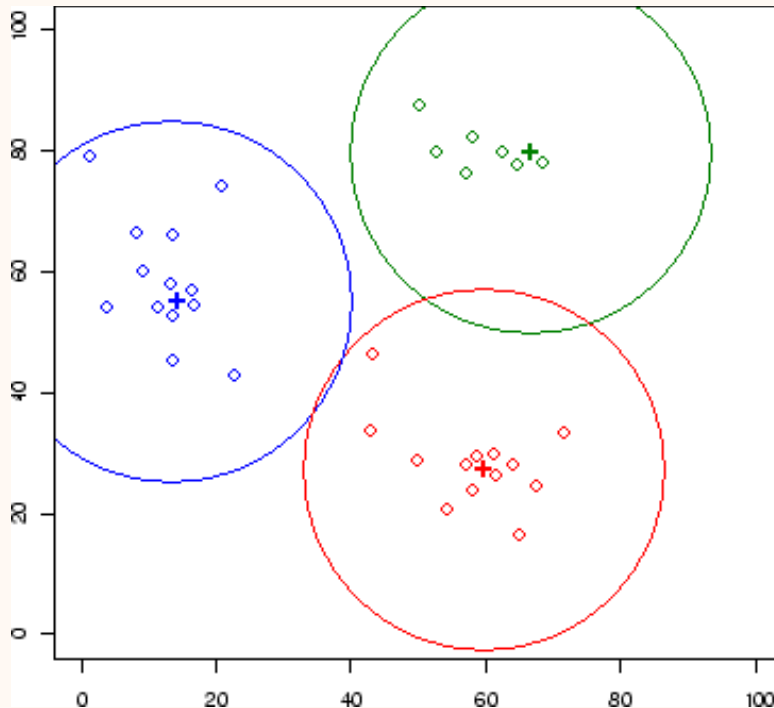
**Clasificación
supervisada**

- **Mínima distancia**

- **Paralelepípedos**

- **Máxima verosimilitud**

ALGORITMO DE MÍNIMA DISTANCIA



$$SD_{xyz} = \sqrt{\sum_{i=1}^n (\mu_{ci} - X_{xyi})^2}$$

Clasificación con mínima distancia

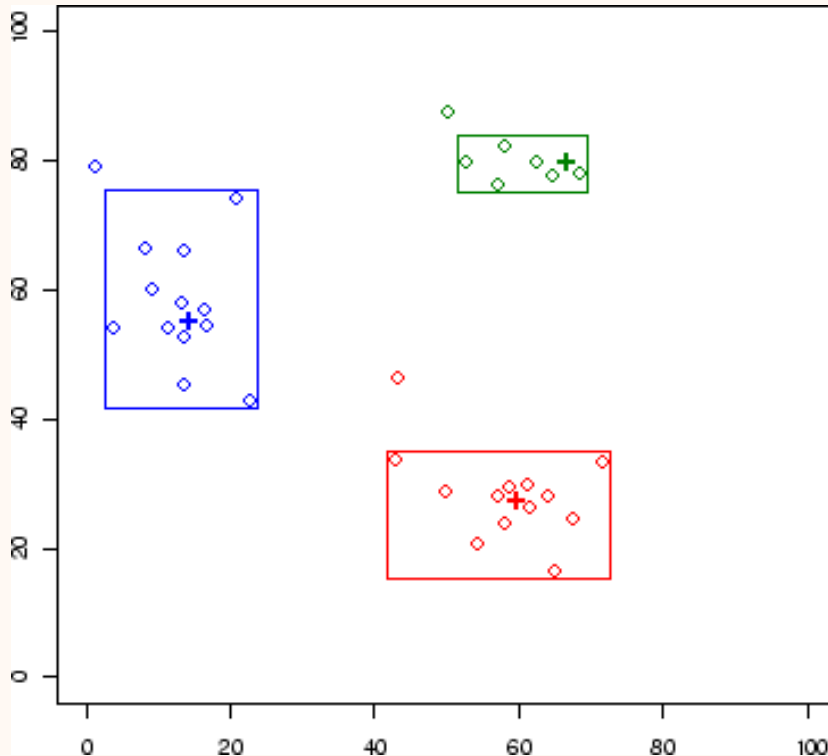
**Sobreclasificación de píxeles
(proclive a errores de comisión)**

**No considera la varianza
(variabilidad de las clases)**



INCONVENIENTES

ALGORITMO DE PARALELEPÍEDOS



Clasificación con paralelepípedos

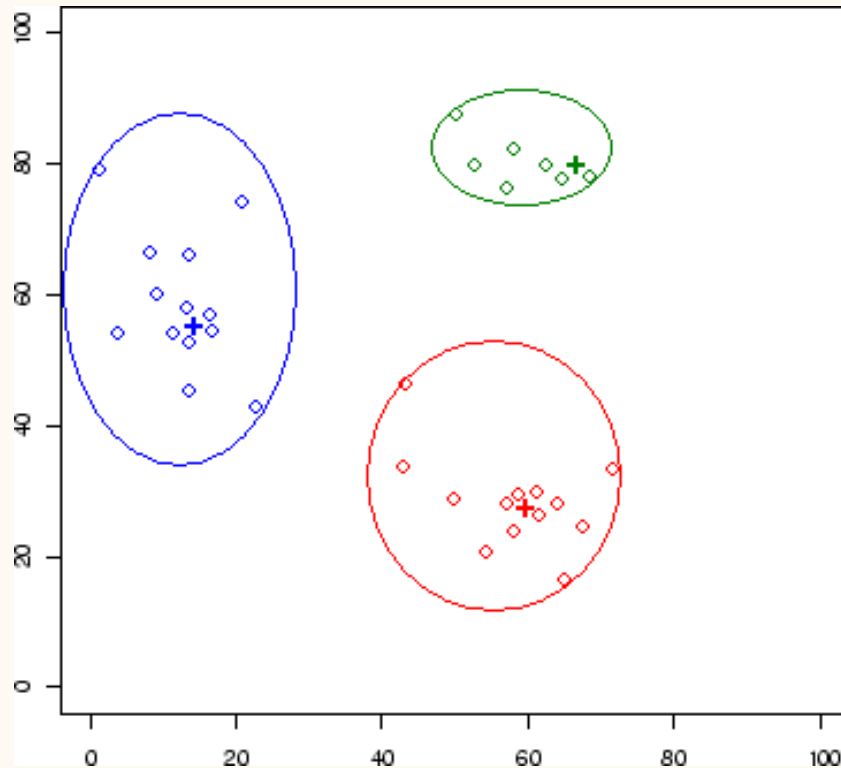
Algunos píxeles quedan sin clasificar

A veces algunos píxeles pueden clasificarse erróneamente en varios grupos



INCONVENIENTES

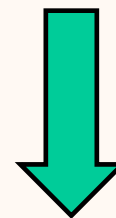
ALGORITMO DE MÁXIMA VEROSIMILITUD



Clasificación con máxima verosimilitud

Gran coste computacional

**Sobreclasificación
de las firmas**



INCONVENIENTES

ALGORITMOS CONTEXTUALES DE POSTCLASIFICACIÓN

- Utilizados sobre todo para postclasificación.
- Mejoran la clasificación clásica espectral usando también información contextual de los vecinos de cada pixel.
- Sobre todo están basados en la fórmula de Bayes.

$$\theta_0 = \omega_r \quad \text{if} \quad p(\xi|\eta_r)P(\omega_r|\tilde{\eta}) = \max_{s=1,\dots,R} p(\xi|\eta_s)P(\omega_s|\tilde{\eta})$$

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

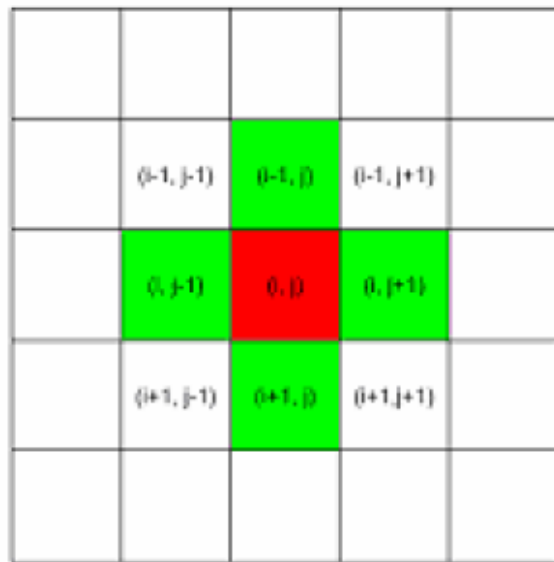
- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

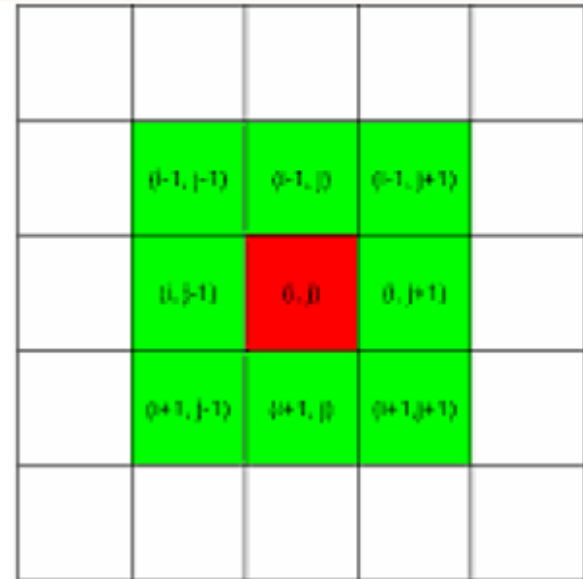
- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

DEFINICION CONCEPTUAL DE AUTOMATA CELULAR

- Autómata celular: modelo matemático para estudiar comportamientos de sistemas complejos (von Neumann, 40s).
- Componentes: celdas, estados, vecindad, reglas y función de transición.



Vecindad de Neumann



Vecindad de Moore

DEFINICION FORMAL DE AUTOMATA CELULAR

Definición (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 célula se expresa mediante un vector de Z^d . Con $d = 1$ tenemos un autómata unidimensional y con posición de las células en Z ; con $d = 2$ tenemos un autómata bidimensional y con posición de las células en $Z \times Z$, etc.*
- r : índice que marca el tamaño de la vecindad, es decir, el número de vecinos que interactúan con la 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.*

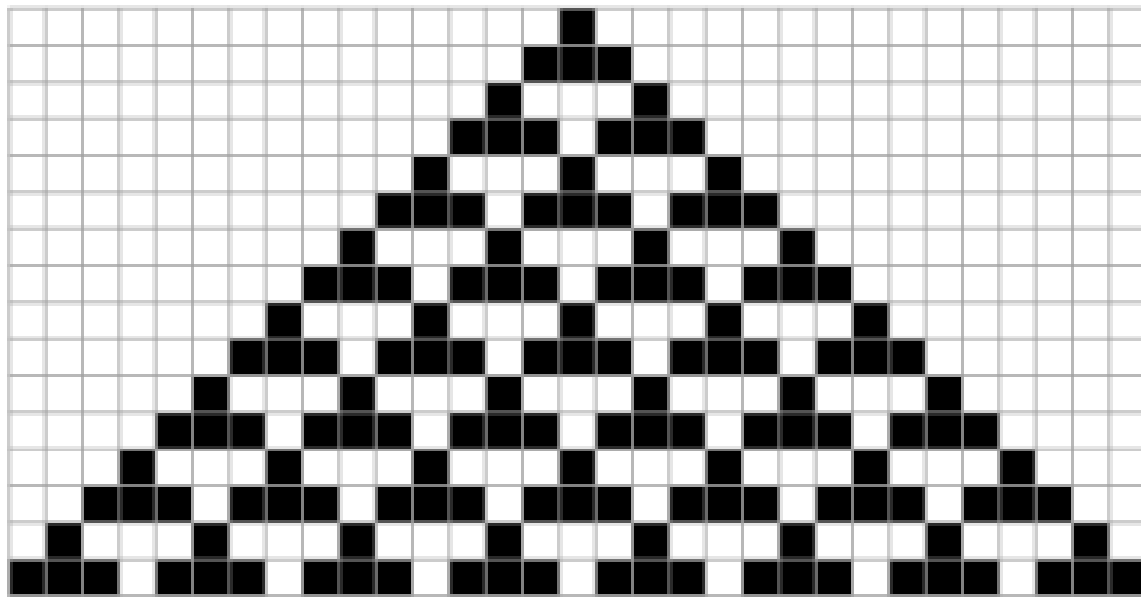
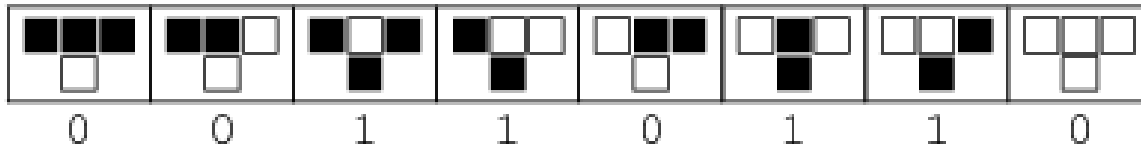
$$f : Q^{r+1} \rightarrow Q$$

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

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

AUTÓMATAS CELULARES UNIDIMENSIONALES

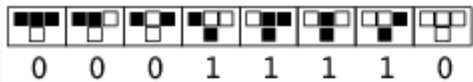
rule 54



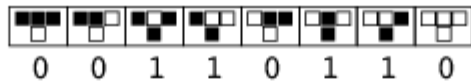
AUTÓMATAS CELULARES

REGLAS MÁS CONOCIDAS DE STEPHEN WOLFRAM

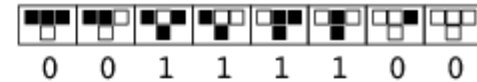
rule 30



rule 54



rule 60



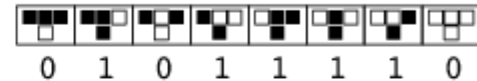
rule 62



rule 90



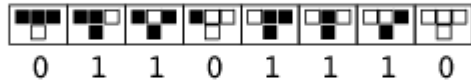
rule 94



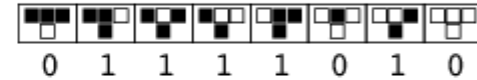
rule 102



rule 110



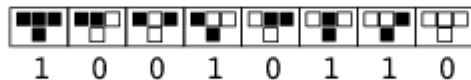
rule 122



rule 126



rule 150



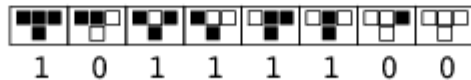
rule 158



rule 182



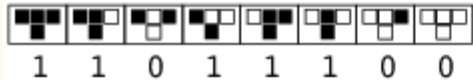
rule 188



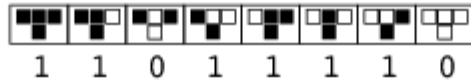
rule 190



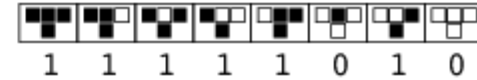
rule 220



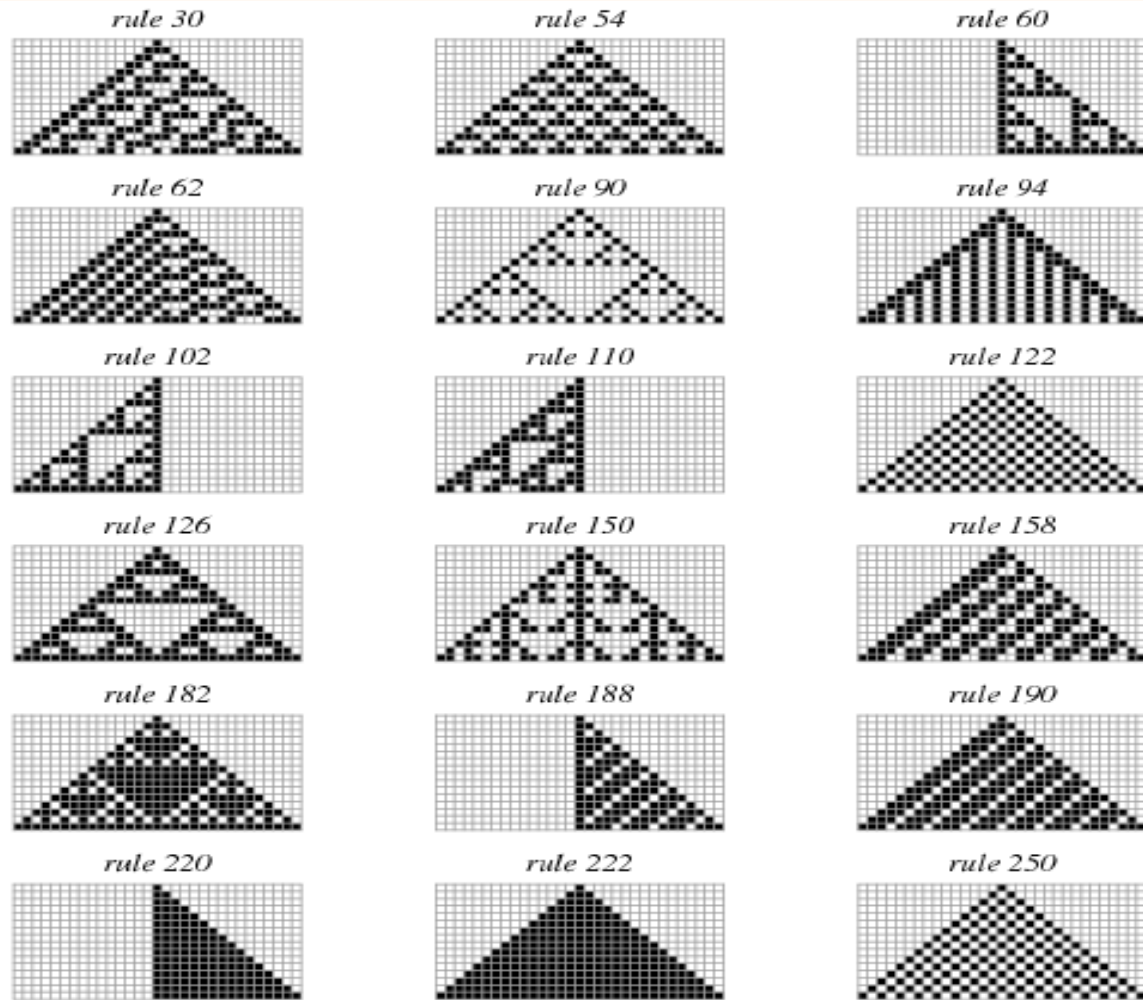
rule 222



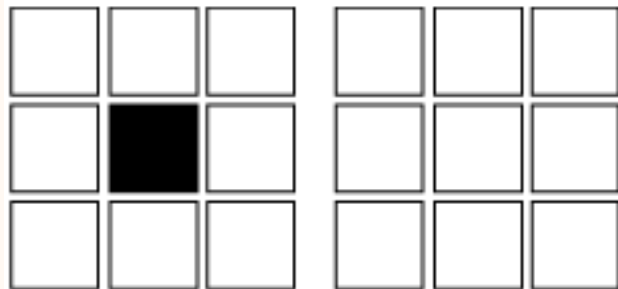
rule 250



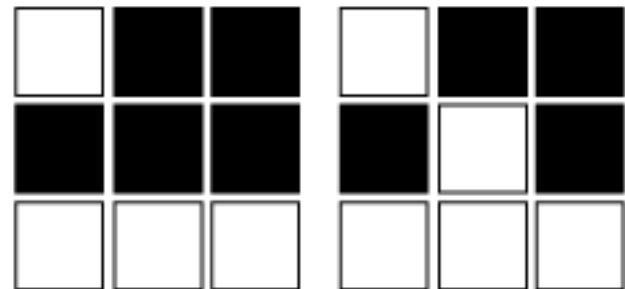
IMÁGENES MÁS CONOCIDAS DE STEPHEN WOLFRAM



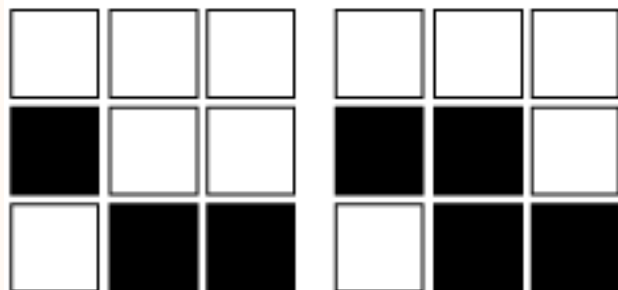
AUTÓMATAS CELULARES BIDIMENSIONALES. JUEGO DE LA VIDA



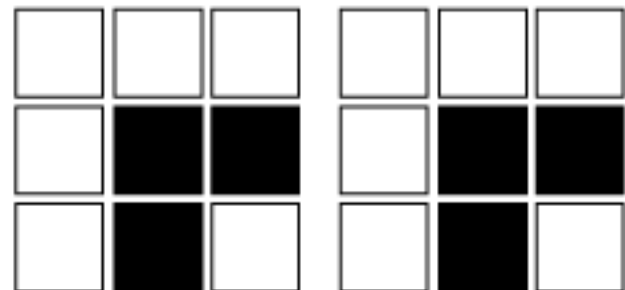
(a)



(b)

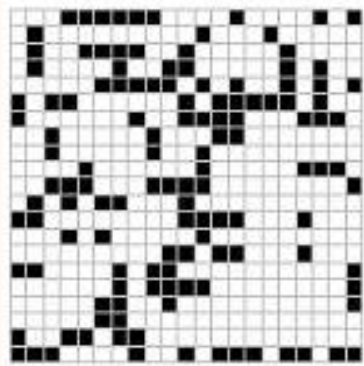


(c)

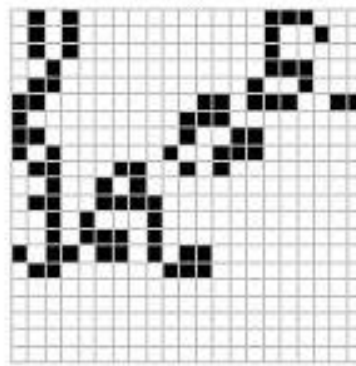


(d)

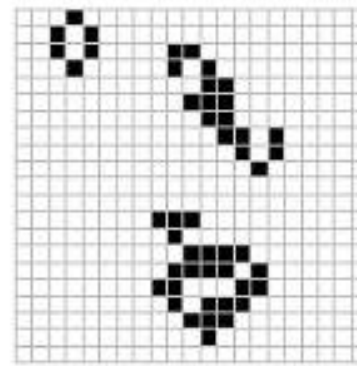
SIMULACIÓN DEL JUEGO DE LA VIDA



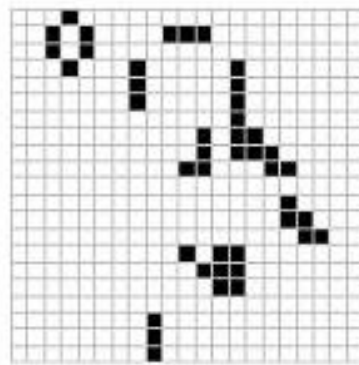
(a)



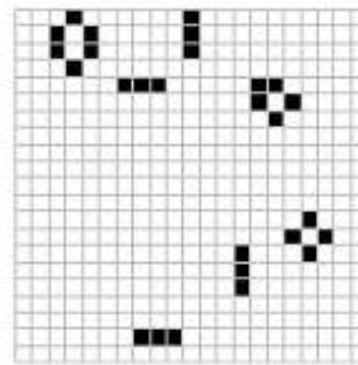
(b)



(c)



(d)



(e)

SIMULACIÓN DE DAÑOS PRODUCIDOS EN VEGETACIÓN POR INUNDACIÓN

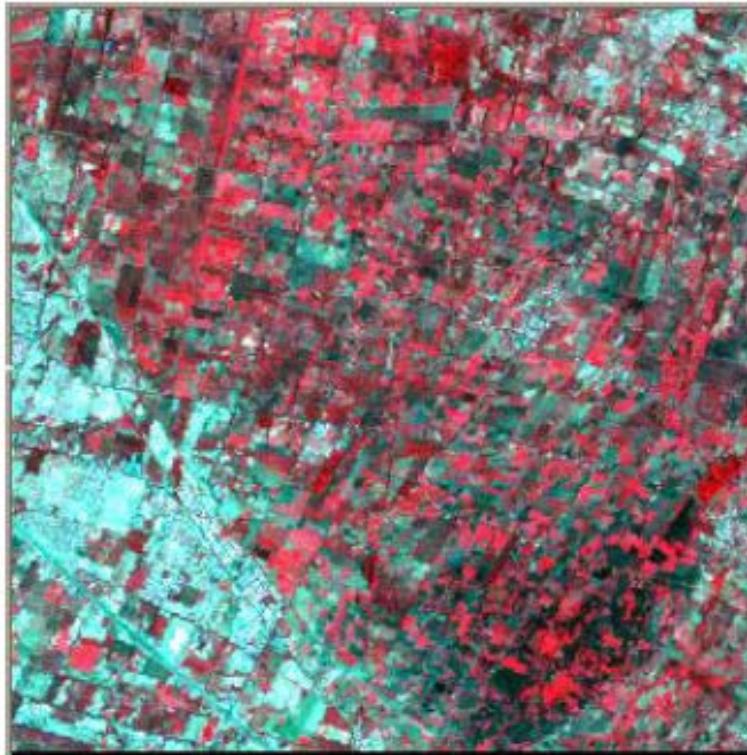


Imagen original

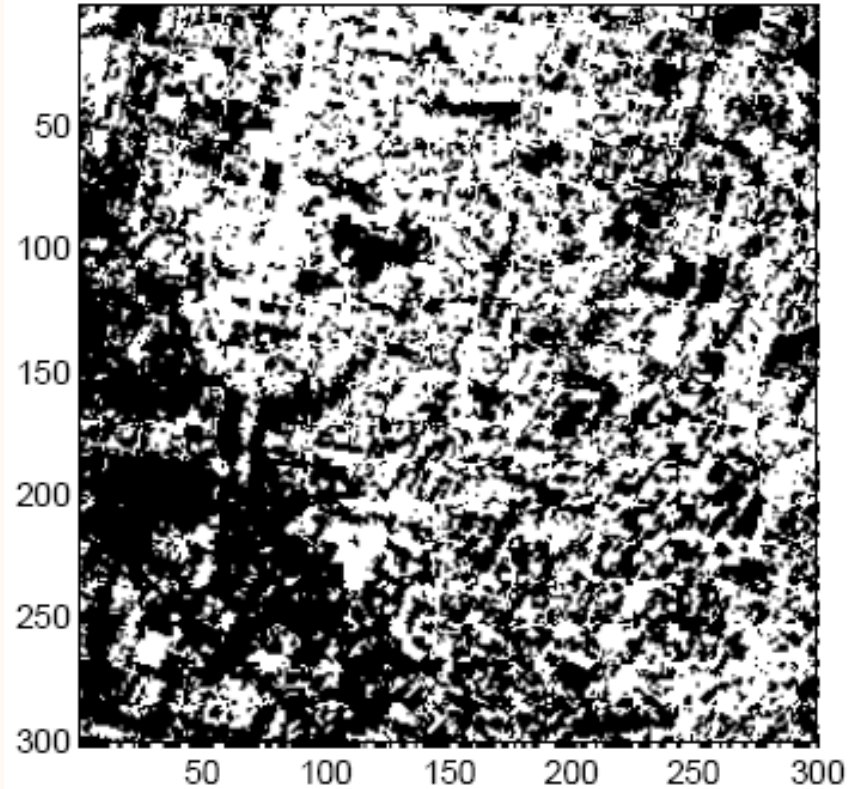


Imagen clasificada

SIMULACIÓN DE NUBES EN IMÁGENES DE SATÉLITE

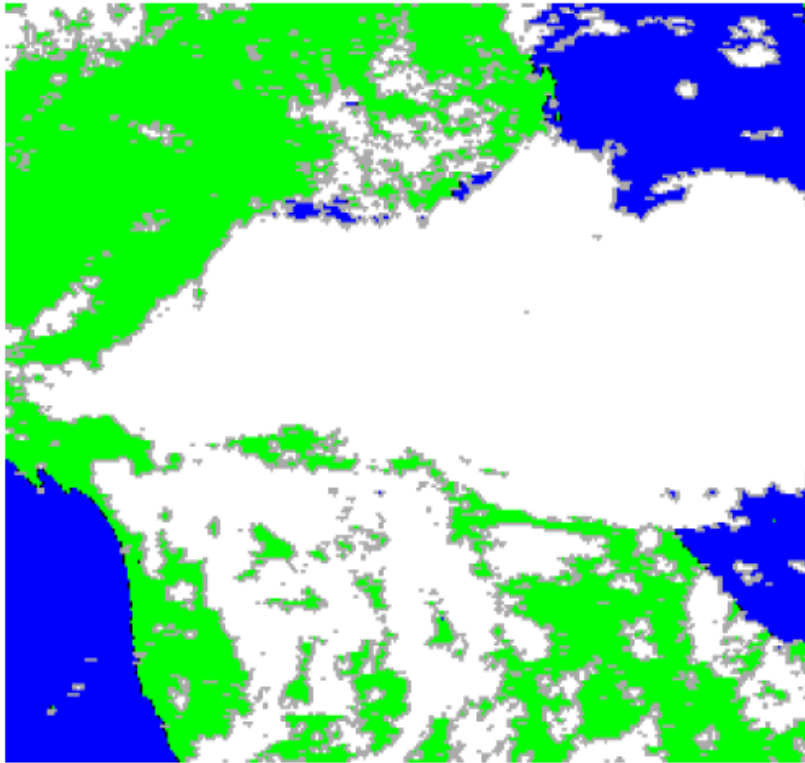


Imagen NOAA14 actual. Clasificación (tierra, mar, nube y borde de nube).

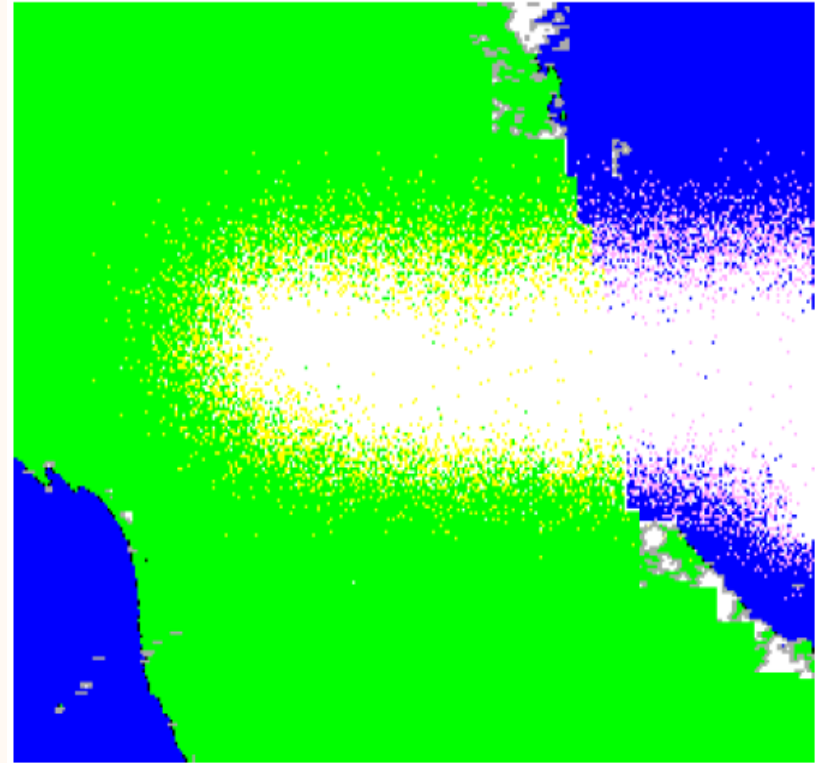
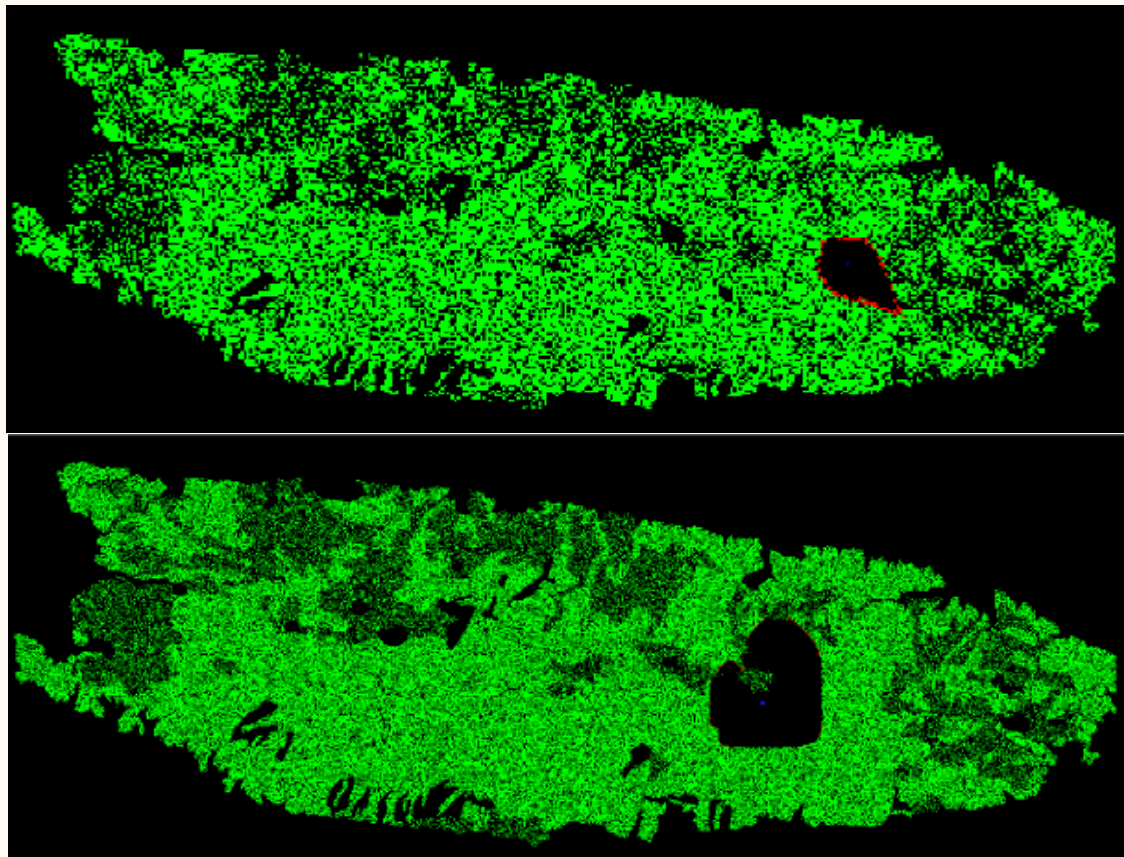


Imagen simulada con AC. 3 fuentes de partículas.

SIMULACIÓN DE PROPAGACIÓN DE INCENDIOS



Zona ardiendo y zona quemada

PROCESAMIENTO DE IMÁGENES



(a) Imagen original, (b) Imagen con ruido, (c) Filtro Gaussiano, (d) Filtro ACs

DETECCIÓN DE BORDES Y FRONTERAS

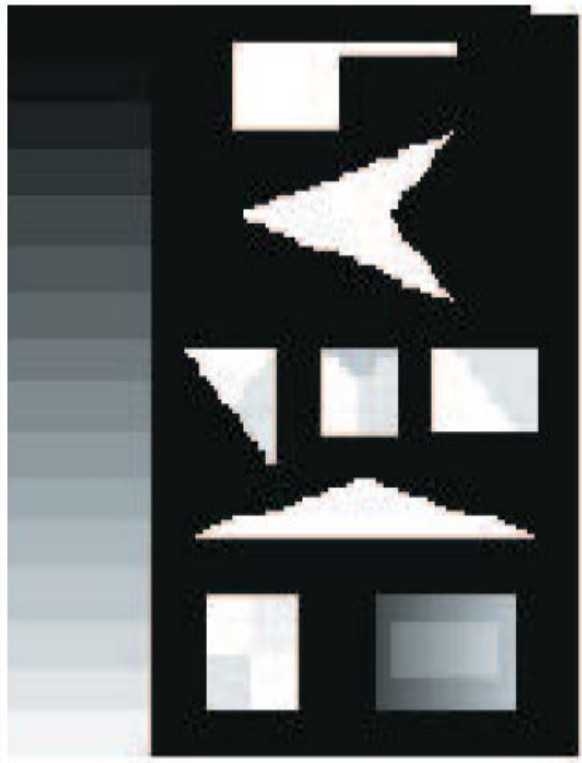
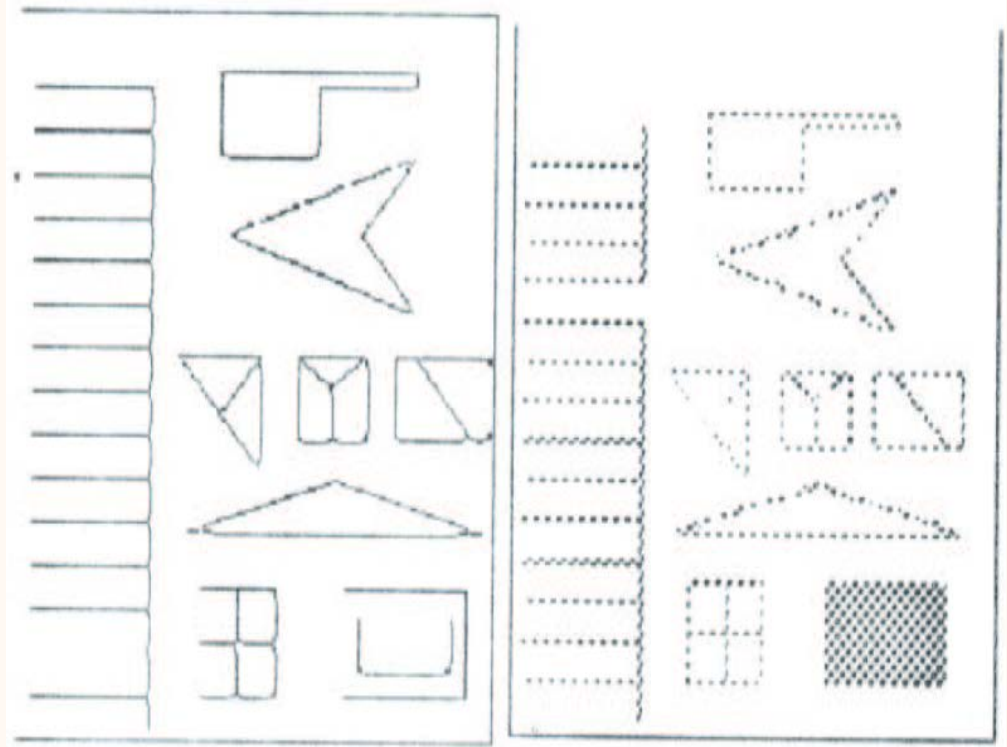


Imagen original



(a) Algoritmo SUSAN, (b) Algoritmos ACs

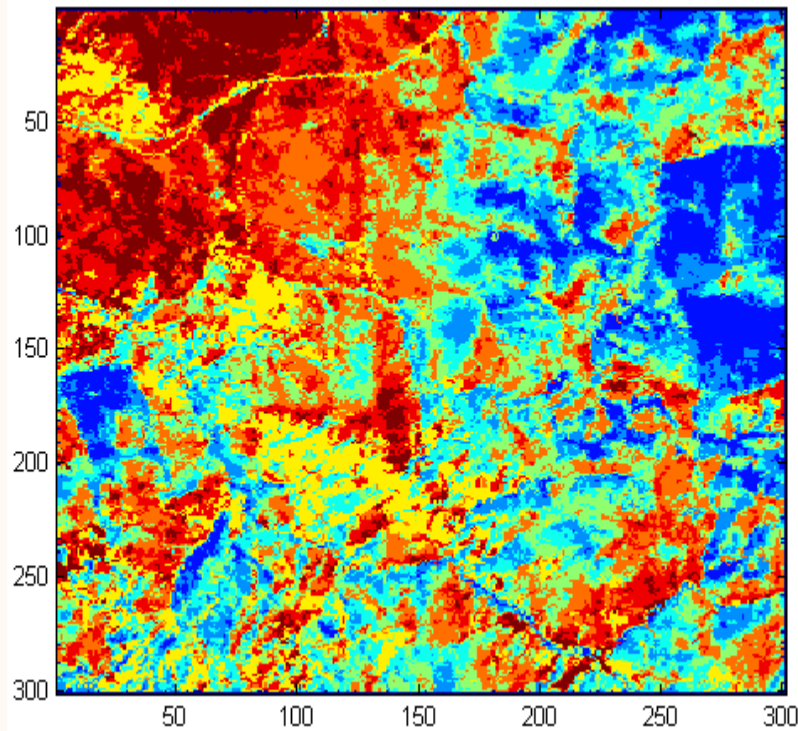
Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

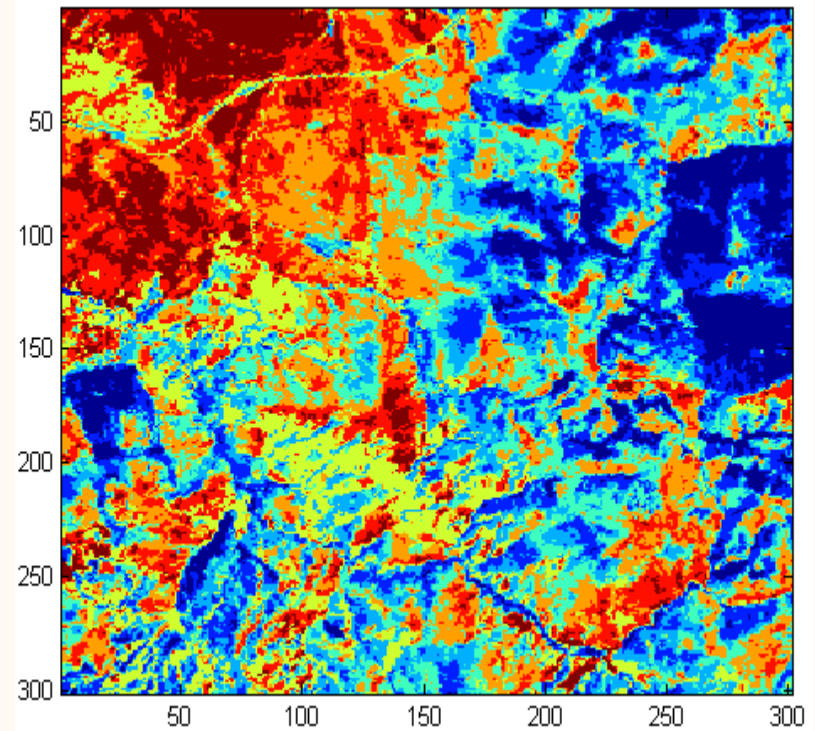
Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

PROBLEMAS DE LOS ALGORITMOS SUPERVISADOS CLÁSICOS



Clasificación con trabajo de campo



Clasificación con mínima distancia

MATRIZ DE CONFUSIÓN (INDICE DE CALIDAD)

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

Matriz de confusión (80% bien clasificado, 20% mal clasificado)

PIXELES INCIERTOS EN EL PROCESO DE CLASIFICACIÓN

Pixel	Clases	Pixel	Clases	Pixel	Clases	Pixel	Clases
[2, 218]	5, 7	[93, 92]	7, 8	[174, 32]	4, 5	[238, 259]	5, 7
[3, 161]	5, 7	[94, 109]	7, 8	[174, 253]	7, 8	[239, 34]	7, 8
[3, 162]	5, 7	[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	[99, 146]	5, 7	[182, 169]	4, 5	[250, 82]	5, 7
[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	[252, 157]	5, 7
[11, 164]	4, 5	[104, 276]	5, 7	[189, 300]	4, 5	[253, 134]	2, 4
[12, 168]	5, 7	[115, 270]	2, 4	[190, 199]	5, 7	[254, 116]	5, 7
[14, 294]	4, 5	[116, 191]	4, 5	[194, 50]	5, 7	[255, 161]	7, 8
[18, 144]	5, 7	[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	[196, 232]	5, 7	[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
[28, 285]	5, 7	[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	[262, 205]	5, 7
[34, 250]	5, 7	[127, 239]	5, 7	[198, 300]	1, 2	[263, 17]	4, 5
[36, 213]	2, 4	[127, 240]	5, 7	[199, 16]	2, 4	[263, 48]	2, 4
[38, 206]	5, 7	[133, 178]	4, 5	[199, 83]	5, 7	[264, 70]	2, 4
[43, 236]	4, 5	[133, 225]	5, 7	[199, 198]	5, 7	[270, 210]	7, 8
[43, 260]	4, 5	[137, 156]	4, 5	[200, 22]	4, 5	[271, 55]	4, 5
[44, 286]	5, 7	[141, 90]	8, 6	[200, 26]	4, 5	[274, 121]	2, 4
[45, 278]	5, 7	[141, 174]	5, 7	[203, 76]	2, 4	[278, 166]	8, 6
[50, 230]	2, 4	[142, 174]	5, 7	[203, 238]	4, 5	[279, 279]	5, 7
[51, 243]	5, 7	[144, 202]	4, 5	[204, 61]	5, 7	[280, 93]	4, 5
[52, 197]	5, 7	[145, 231]	5, 7	[205, 18]	2, 4	[281, 45]	2, 4
[55, 52]	7, 8	[148, 134]	7, 8	[205, 59]	4, 5	[282, 226]	8, 6

ANÁLISIS DE LOS PROBLEMAS DE CLASIFICACIÓN CLÁSICA

ImagInfo (almeriagranadagrandetrozoiso.data.img)

File Edit View Help

1 Layer 1

General Projection Histogram Pixel data

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	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	8	8	8	8	8	8	8	7	5	4	4	5	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	7	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	9	7	5	4	4	7	8
22	7	7	5	7	7	7	7	8	8	8	8	8	8	7	7	7	7	7	8

Clasificación con trabajo de campo

ImagInfo (almeriagranadagrandetrozomindist.img)

File Edit View Help

1 Layer 1

General Projection Histogram Pixel data Classification_Info

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	8	8	8	8	8	8	8	7	5	4	4	5	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	7	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	9	7	5	4	4	7	8
22	7	7	5	7	7	7	7	8	8	8	8	8	8	7	7	7	7	7	8

Clasificación con mínima distancia

ANÁLISIS DE LOS PROBLEMAS DE CLASIFICACIÓN CLÁSICA

[2, 218], (5, 7)	Clase 1
[3, 161], (5, 7)	-----
[3, 162], (5, 7)	Capa 1: 56.7658
[3, 293], (4, 5)	Capa 2: 25.5007
[6, 241], (1, 2)	Capa 3: 28.0801
[7, 9], (5, 3)	Capa 4: 48.5983
[7, 245], (1, 2)	Capa 5: 53.8074
[8, 206], (2, 4)	Capa 6: 142.911
[8, 207], (2, 4)	Capa 7: 24.5118
[8, 271], (2, 4)	
[10, 190], (2, 4)	Clase 2
[11, 164], (4, 5)	-----
[12, 168], (5, 7)	Capa 1: 65.6057
[14, 294], (4, 5)	Capa 2: 31.5801
[18, 144], (5, 7)	Capa 3: 38.1911
[20, 169], (4, 5)	Capa 4: 53.6608
[21, 169], (4, 5)	Capa 5: 74.2405
[26, 145], (7, 8)	Capa 6: 150.921
[27, 251], (4, 5)	Capa 7: 36.6283
[28, 285], (5, 7)	
[29, 174], (4, 5)	Clase 3
[34, 230], (2, 4)	-----
[34, 250], (5, 7)	Capa 1: 91.8313
[36, 213], (2, 4)	Capa 2: 46.3399
[38, 206], (5, 7)	Capa 3: 59.8819
[43, 236], (4, 5)	Capa 4: 62.9785
[43, 260], (4, 5)	Capa 5: 96.1951
[44, 286], (5, 7)	Capa 6: 167.733
[45, 278], (5, 7)	Capa 7: 51.7341
[50, 230], (2, 4)	

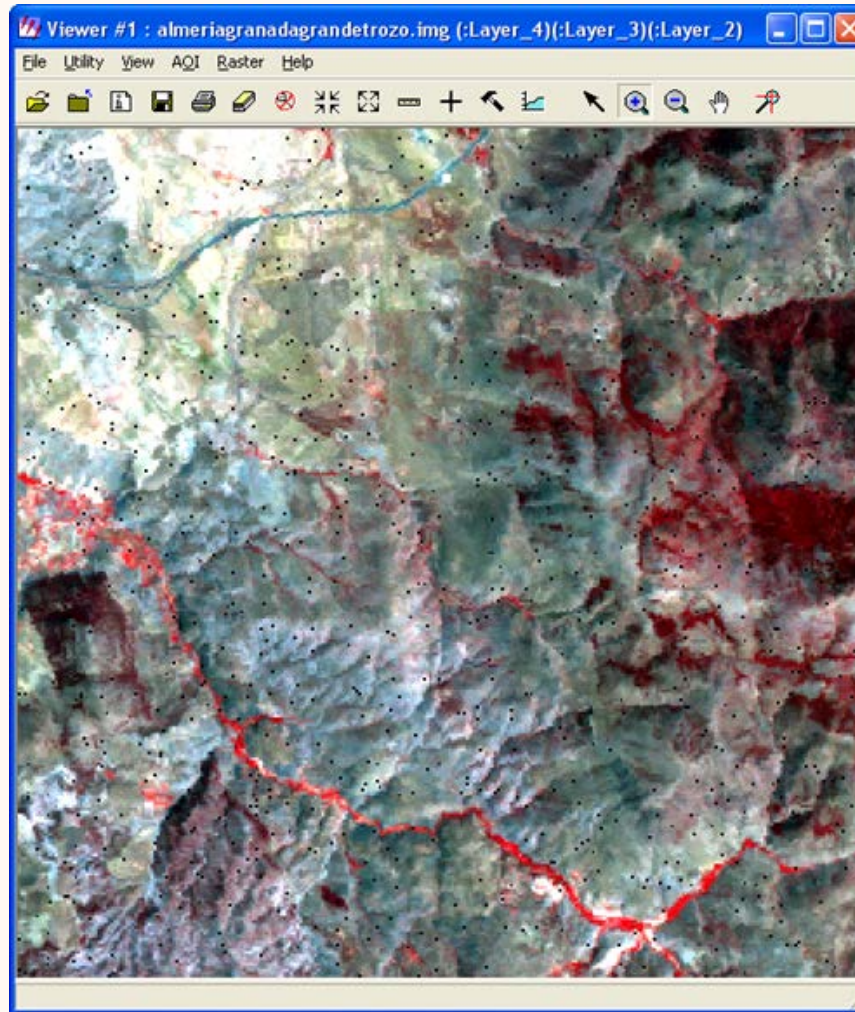
Row	158	159	160	161	162	163	164	165	166	167	168	169	170
0	34	70	46	35	35	33	46	54	46	58	67	66	63
1	34	61	51	44	44	47	53	45	47	66	67	67	67
2	43	40	55	52	52	56	47	44	54	66	65	74	69
3	65	60	49	54	54	54	46	52	64	63	70	70	70
4	73	68	62	51	49	41	61	72	64	59	67	68	71
5	73	67	47	45	50	56	62	64	63	63	74	74	75
6	62	55	47	47	60	38	69	61	54	54	67	72	69
7	55	46	47	63	53	35	34	30	42	56	59	72	70
8	49	52	54	66	52	38	38	31	41	54	53	74	72
9	46	55	59	55	60	59	42	41	42	50	49	71	72
10	46	47	53	54	54	53	44	45	47	53	48	66	73
11	54	55	51	58	58	53	45	45	52	55	67	72	73
12	66	62	56	60	64	58	49	45	45	50	62	75	76
13	67	68	64	66	74	74	72	52	61	45	50	70	66
14	68	70	69	72	71	81	76	61	65	50	40	55	68
15	70	71	69	70	73	76	87	63	54	51	34	46	66
16	70	71	70	70	74	82	75	65	55	40	40	48	59
17	71	73	72	72	70	73	70	68	58	47	49	52	60
18	66	73	69	70	71	71	72	68	59	54	47	43	60
19	66	73	74	73	70	73	74	71	63	54	42	48	58
20	63	73	74	73	70	73	74	73	62	50	43	53	62
21	71	73	70	70	71	72	71	73	62	50	43	53	68
22	68	76	73	70	73	73	67	71	61	59	63	62	70

Píxeles erróneos

Archivo SIG

Capas de la imagen original

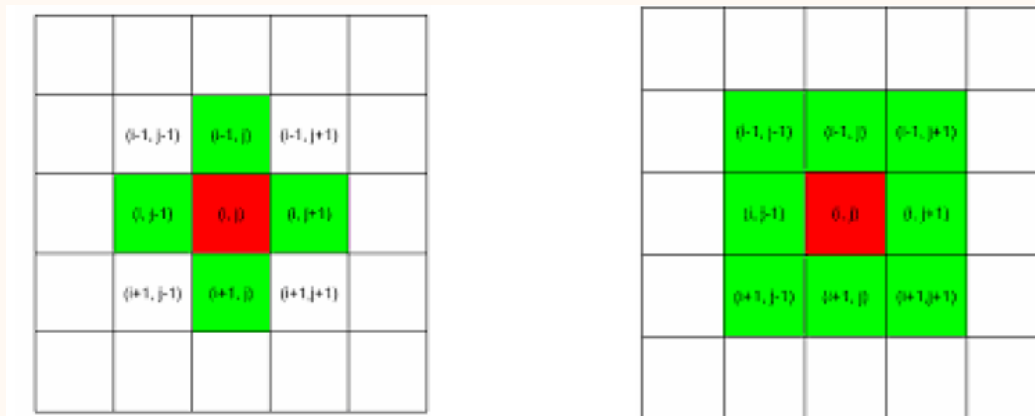
PÍXELES RUIDOSOS



OBJETIVOS DEL ALGORITMO ACA

OBJETIVOS DE ACA

1. Píxeles inciertos: si está próximo a dos clases, tomar la clase mayoritaria de alrededor (VECINDAD).
2. Eliminación del ruido: píxeles con ruido impulsivo (VECINDAD).
3. Clasificación iterativa en niveles de fiabilidad (ITERACIONES).



DEFINICIÓN DEL AUTÓMATA CELULAR DE ACA

CONSIDERACIONES INICIALES

- Cada pixel de la imagen corresponde con una celda del AC.
- Cada clase de la clasificación corresponde a un estado de las celdas.
- La vecindad de cada celda corresponde con los 4, 8 ó 24 pixeles vecinos.
- La función de transición (f) debe clasificar correctamente cada pixel de la imagen, usando una fusión de clasificación espectral-contextual.

FILTRO DE MAYORÍA

MAJORITY FILTER

- Filtro lógico: reetiqueta el pixel central.
- Se usa la etiqueta de la clase mayoritaria.

If $(n_i > n_j \ \&\& \ n_i > n_t \ \text{for all } i \neq j)$ then $x \in \omega_j$

x = centre pixel,

n_i & n_j = the number of adjacent pixels belong to class i and j

n_t = threshold

DEFINICIÓN DE LOS ESTADOS DEL AUTÓMATA CELULAR DE ACA

ESTADOS DEL AUTÓMATA CELULAR

AC multiestado: array de estados posibles [clase] [tipo][iteración]

- Clase: clases del .sig.
 - Tipo: núcleo, frontera, ruidoso, caótico.
 - Iteración: grado de certidumbre en la clasificación de dicho píxel.
-
- Cto. de estados 1: clases del cto de entrenamiento, clase ruido, clase vacía.
 - Cto. de estados 2: núcleo foco, frontera, caótico, ruidoso.
 - Cto. de estados 3: $1..k$, siendo k el número de iteraciones del AC.

DEFINICIÓN DE LOS ESTADOS DEL AUTÓMATA CELULAR DE ACA

VALORES DE ENTRADA DEL AUTÓMATA CELULAR

Clases de la vecindad, clasificación espectral del píxel actual, k

, donde k es el número actual de iteración,
y la clasificación espectral del píxel actual es:

- Píxel bueno: clase a la que pertenece (si la distancia respecto a las otras es suficiente).
- Píxel caótico: clases dudosas (si las distancias son parecidas).
- Píxel ruidoso: clase ruido (para solventar ruido).

DEFINICIÓN DE LAS REGLAS DEL AUTÓMATA CELULAR DE ACA

REGLAS DEL AUTÓMATA CELULAR

Si píxel bueno y está rodeado de vecinos de clase vacía o de la misma clase:

$$[\text{clase}][\text{tipo}][\text{iteracion}] = \{\text{clase píxel, núcleo de foco, } k\}$$

Si píxel bueno y tiene píxeles alrededor de distinta clase a la suya:

$$[\text{clase}][\text{tipo}][\text{iteracion}] = \{\text{clase píxel, frontera, } k\}$$

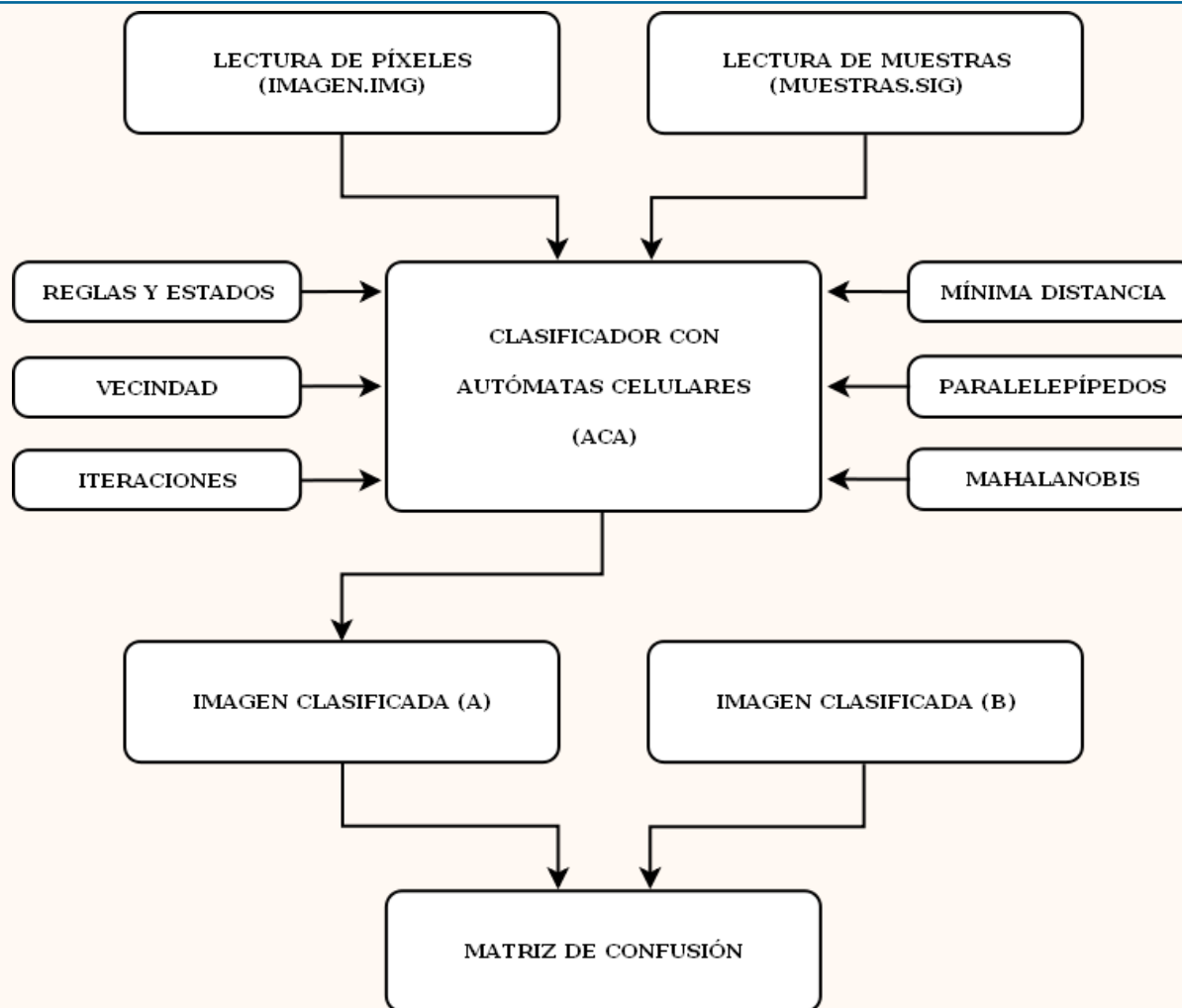
Si píxel caótico:

$$[\text{clase}][\text{tipo}][\text{iteracion}] = \{\text{clase mayoritaria de los vecinos de entre las clases dudosas, caótico, } k\}$$

Si píxel ruidoso:

$$[\text{clase}][\text{tipo}][\text{iteracion}] = \{\text{clase mayoritaria de los vecinos, ruidoso, } k\}$$

ARQUITECTURA COMPLETA DEL ALGORITMO ACA



PSEUDOCÓDIO DEL ALGORITMO ACA PRINCIPAL

Algoritmo ACA ($E, k, numIterations, threshold$)

Entrada:

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

k : 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          $r1 \leftarrow spectralACA(e_i)$ ;
05         if  $r1 \neq \emptyset$  then
06            $r2 \leftarrow contextualACA(e_i, r1)$ ;
07         endif
08       endif
09     end
10      $threshold \leftarrow threshold + incremental$ ;
11   end
```

PSEUDOCÓDIO DEL ALGORITMO ACA ESPECTRAL

Algoritmo spectralACA (e)

Entrada:

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

Salida:

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

```
01   $res1 \leftarrow \emptyset$ ;  
02  for  $i \leftarrow 0$  to  $numClasses$  do  
03    if  $minDistance(e_i, c_j) \mid j \in \{1..k\} \leq threshold$  then  
04       $res1 \leftarrow res1 \cup class_i$ ;  
05    endif  
06  end
```

PSEUDOCÓDIO DEL ALGORITMO ACA CONTEXTUAL

Algoritmo contextualACA (e)

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

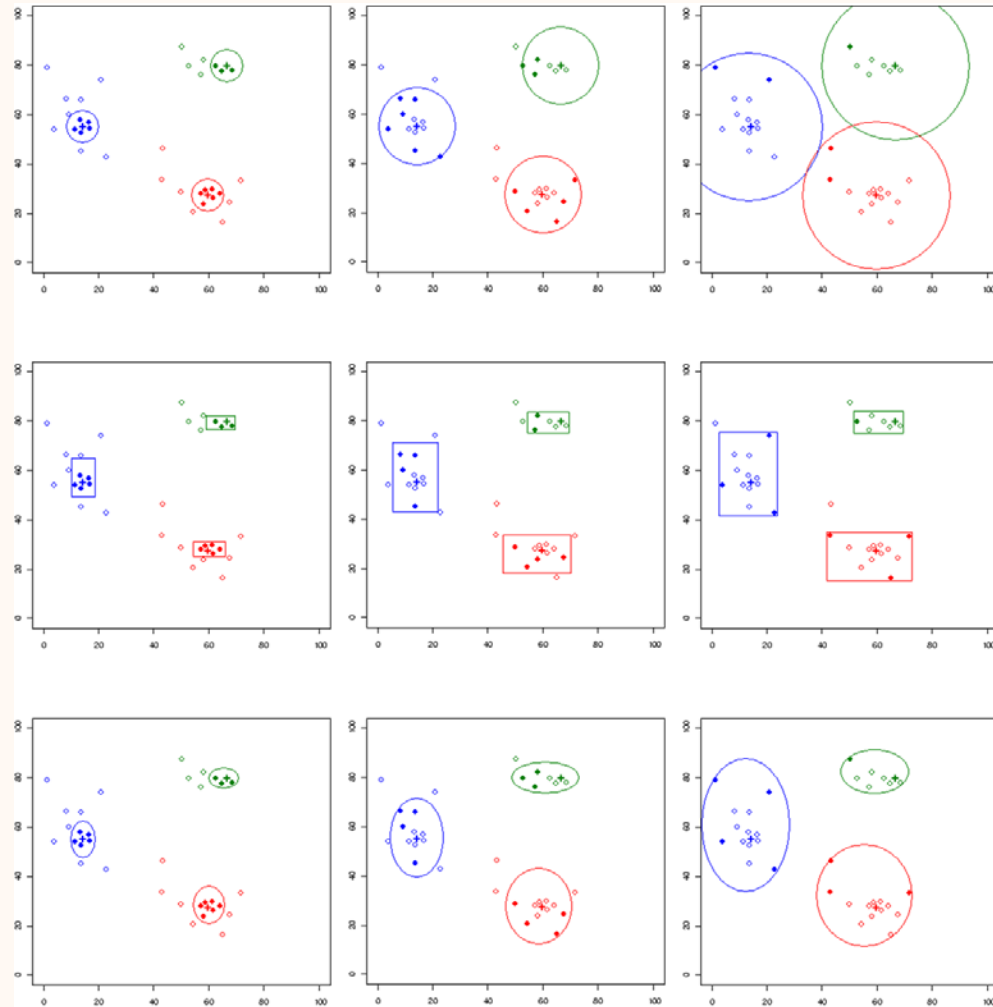
Salida:
 $C = \{c_1, c_2, \dots, c_k\}$: conjunto de centros de las clases

```

01  ifsize(res1) = 1 then
02      if{res1} ≠ noiseClass then
03           $e_i.class \leftarrow \{res1\}$ ;
04           $e_i.quality \leftarrow numIteration$ ;
05          ifneighbourhoodClassesType( $e_i$ ) = 1 then
06               $e_i.type \leftarrow focusPixel$ ;
07          endif
08          ifneighbourhoodClassesType( $e_i$ ) = 2 then
09               $e_i.type \leftarrow edgePixel$ ;
10          endif
11      endif
12      if{res1} = 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  ifsize(res1) ≠ 1 then
19       $e_i.class \leftarrow bayesNeighbourhoodClass()$ ;
20       $e_i.quality \leftarrow numIteration$ ;
21       $e_i.type \leftarrow caoticPixel$ ;
22  endif

```

SIMULACIÓN DEL PROCESO DE CLASIFICACIÓN CON ACA



Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

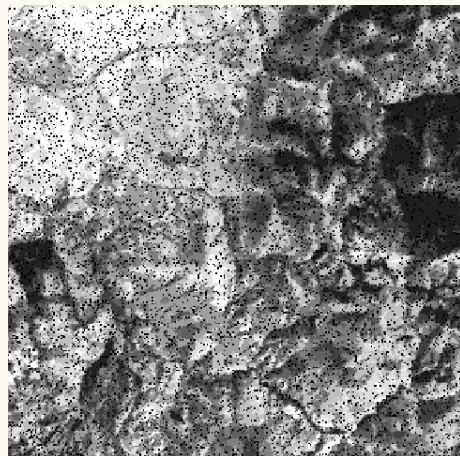
Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

MATRIZ DE CONFUSIÓN DEL ALGORITMO DE MÍNIMA DISTANCIA



Clasificación con trabajo de campo



Clasificación con Mínima Distancia

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

Matriz de confusión del algoritmo mínima distancia
Arriba: Campo. Izquierda: Mínima distancia. (8 clases)

Píxeles bien clasificados: $72.974 / 90.601 = 80\%$

MATRIZ DE CONFUSIÓN DEL ALGORITMO ACA



Clasificación con trabajo de campo



Clasificación con ACA

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

Matriz de confusión del algoritmo ACA
Arriba: Campo. Izquierda: ACA. (8 clases)

Píxeles bien clasificados: $75.899 / 90.601 = 84\%$

COMPARACIÓN DE MÍNIMA DISTANCIA Y ACA

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

Píxeles bien clasificados:
 $72.974 / 90.601 = 80\%$

Píxeles bien clasificados:
 $75.899 / 90.601 = 84\%$

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

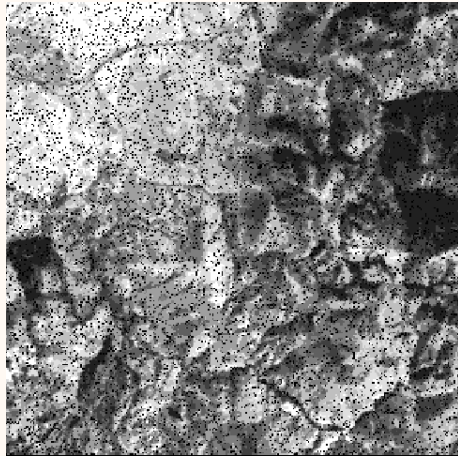
ACA Mejora un 4% los resultados
¡OJO: mejorable!

Matriz de confusión del algoritmo ACA

CLASIFICADOR DE MÍNIMA DISTANCIA VS CLASIFICADOR ACA



**Imagen original
(7 capas, 300x300 p, 10% ruido)**



Clasificación con Mínima Distancia

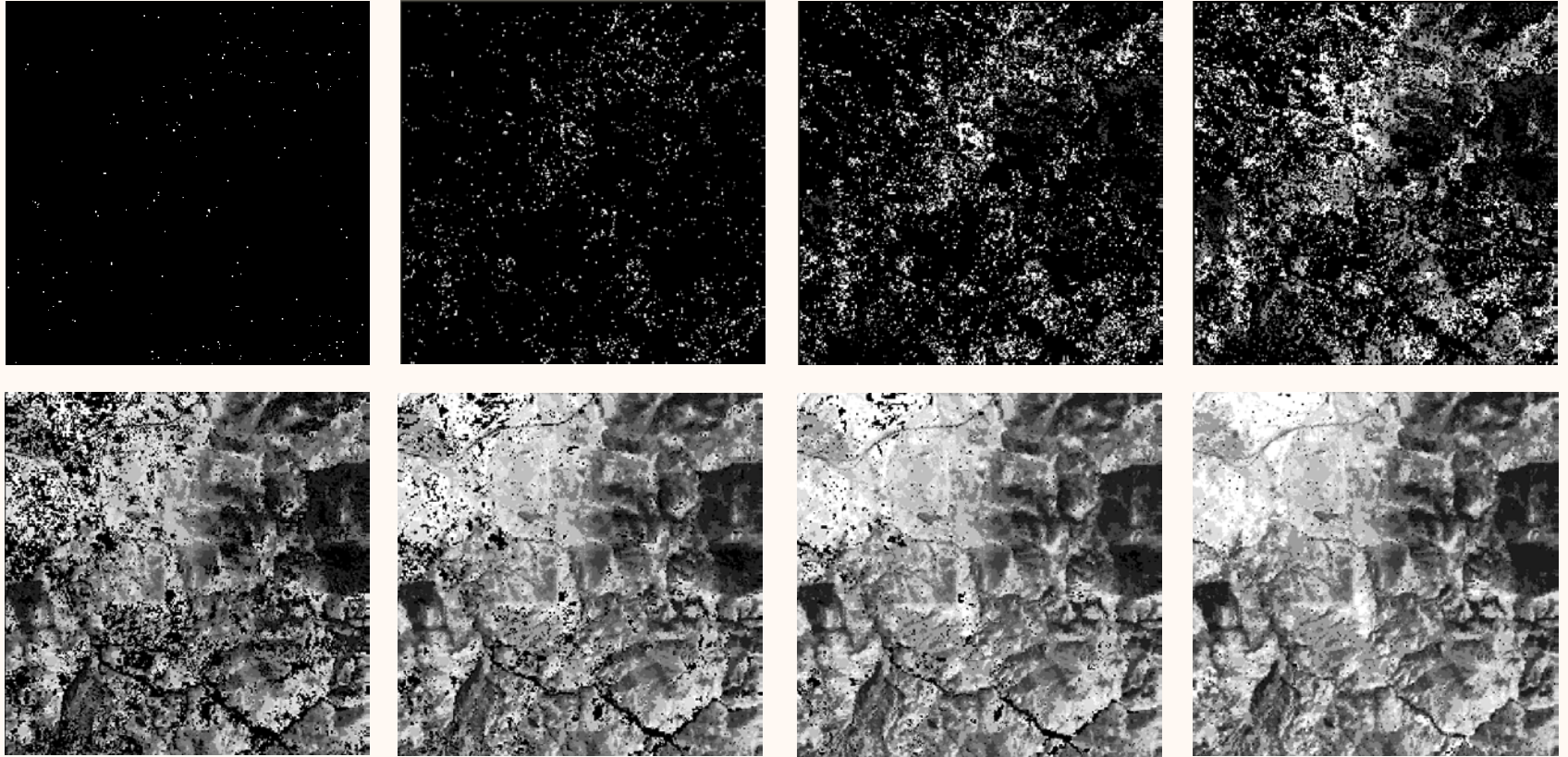


Clasificación con ACA

CLASIFICACIÓN ITERATIVA

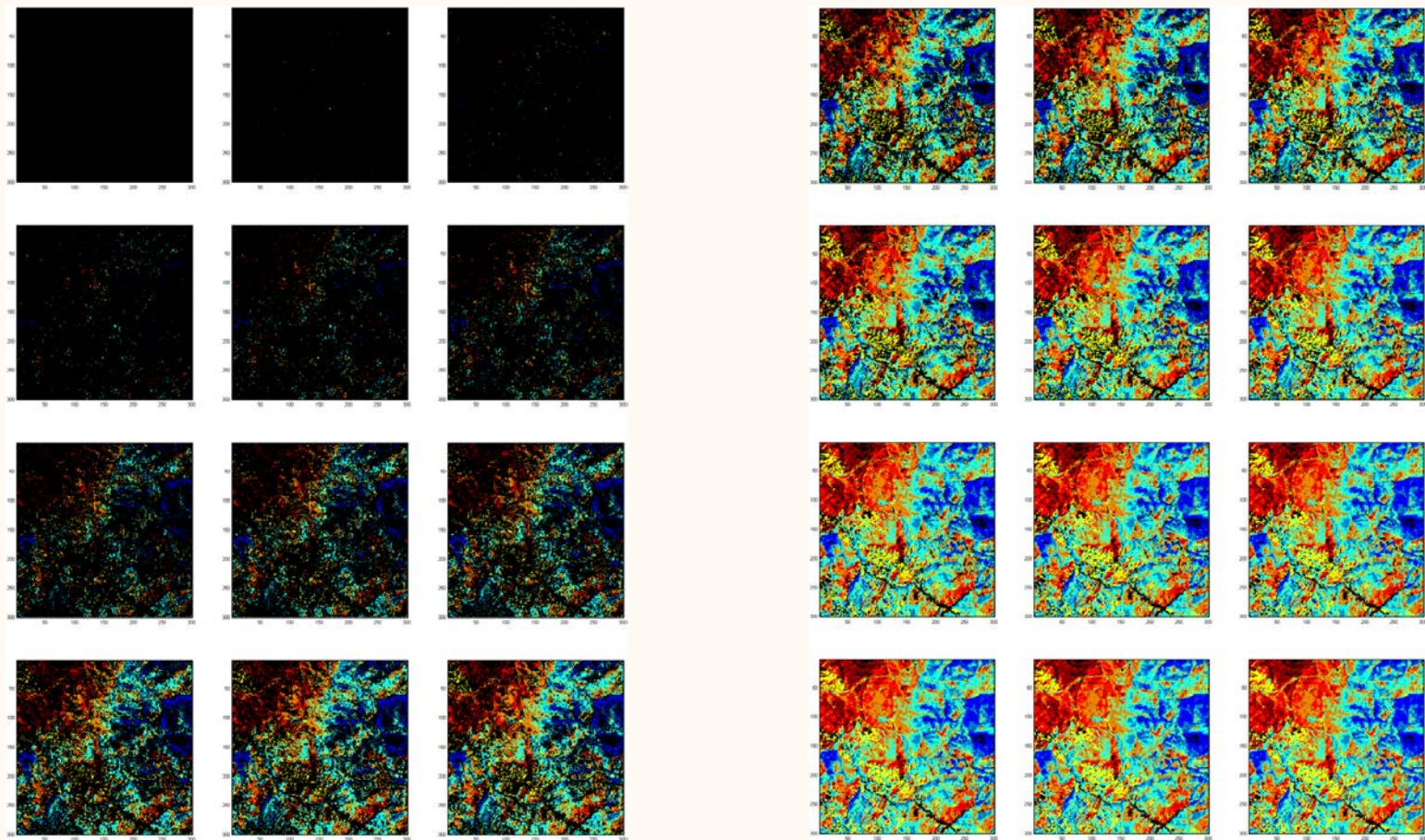


CLASIFICACIÓN JERARQUIZADA EN 8 NIVELES DE FIABILIDAD



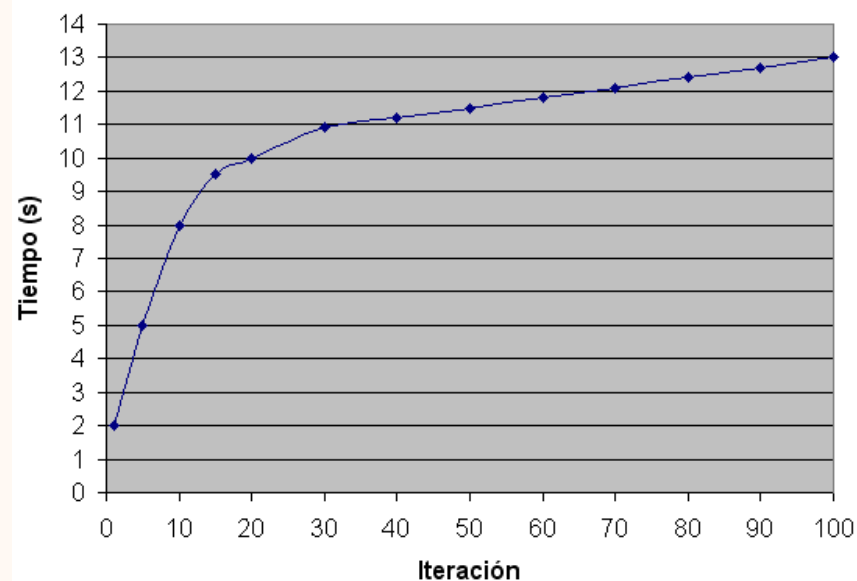
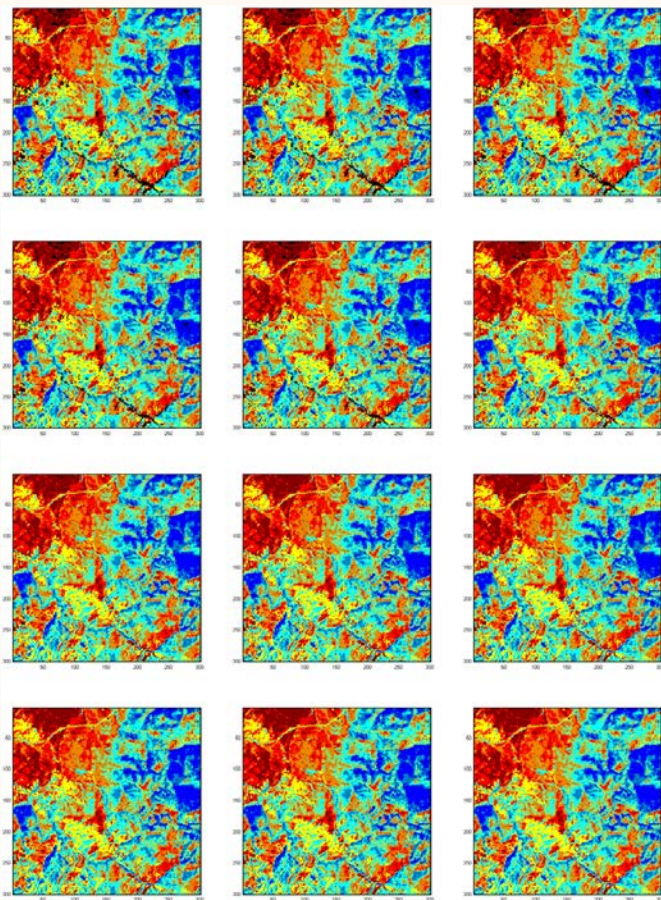
Clasificación jerarquizada en 8 iteraciones del AC

CLASIFICACIÓN JERARQUIZADA EN 100 NIVELES DE FIABILIDAD



Clasificación jerarquizada en 100 iteraciones del AC

CLASIFICACIÓN JERARQUIZADA EN 100 NIVELES DE FIABILIDAD



Clasificación jerarquizada en 100 iteraciones del AC

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

FUTURAS MEJORAS DEL ALGORITMO ACA

TRABAJOS FUTUROS

- **Desarrollo de nuevas versiones del algoritmo de ACA con una configuración de estados y reglas del autómata celular distintas para personalizar aun mas el proceso de clasificación.**
- **Utilizar mejora no sólo en clasificación contextual sino también en clasificación textural.**
- **Utilización de agentes de software para reducir el coste computacional, recorriendo varias regiones de la imagen en paralelo.**
- **Creación de un plugin para la herramienta Erdas Imagine que permita una clasificación personalizada basada en autómatas celulares.**

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

Autómatas celulares multiestado aplicados a la clasificación contextual iterativa

- **TELEDETECCIÓN**
- **AUTÓMATAS CELULARES**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS**
- **PREGUNTAS**

RUEGOS Y PREGUNTAS

AGRADECIMIENTOS

**¡MUCHAS GRACIAS
POR LA ATENCION!**

