

# **CLASIFICACIÓN DE IMÁGENES DE SATÉLITE MEDIANTE AUTÓMATAS CELULARES**

**Antonio Moisés Espínola Pérez**

**- PROYECTO SOLERES -**

15 de febrero de 2010

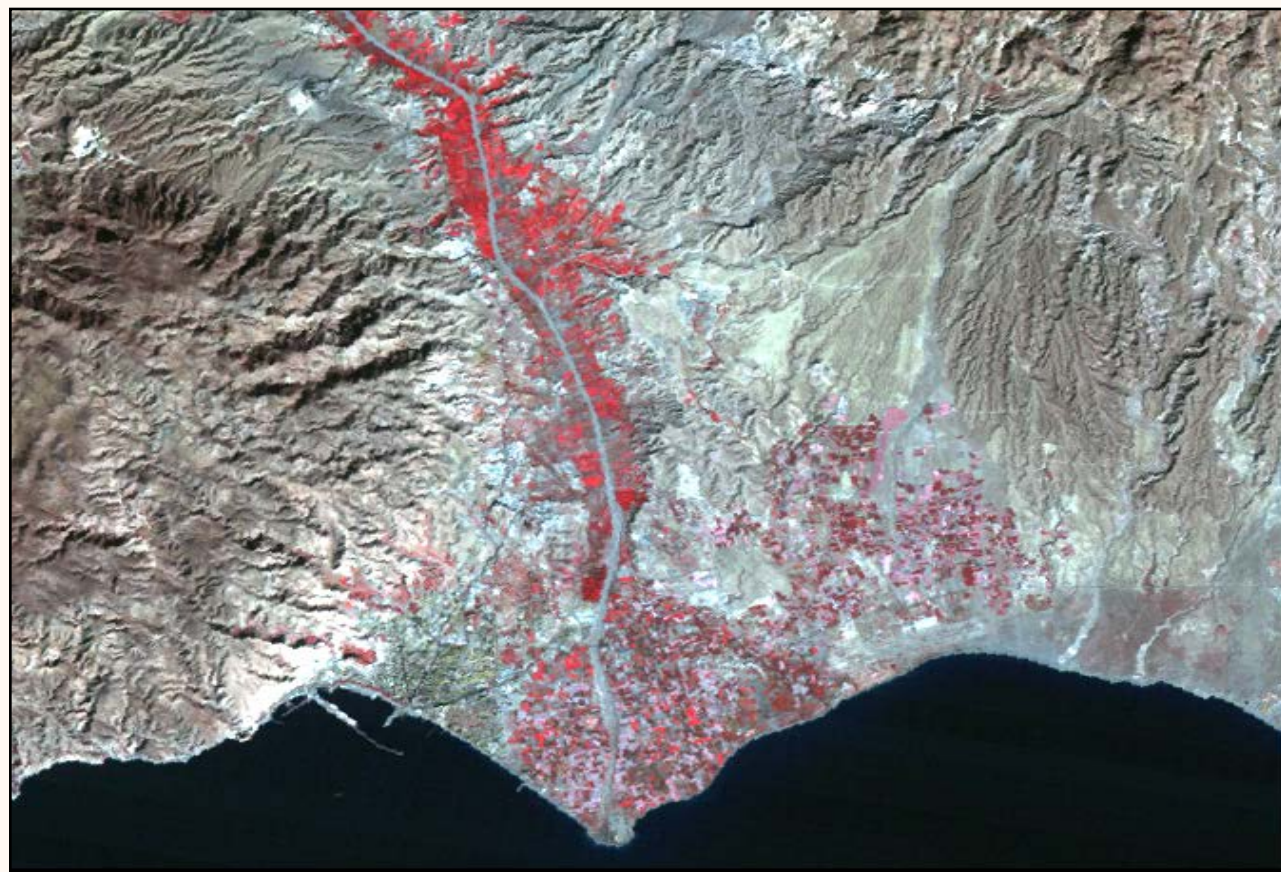
## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## TELEDETECCIÓN



*Imagen de la provincia de Almería*

Clasificación de imágenes de Satélite mediante Autómatas Celulares

## ALGORITMOS DE CLASIFICACIÓN

**Clasificación  
no supervisada**

- K-medias

- ISODATA

**Clasificación  
supervisada**

- Mínima distancia

- Paralelepípedos

- Mahalanobis

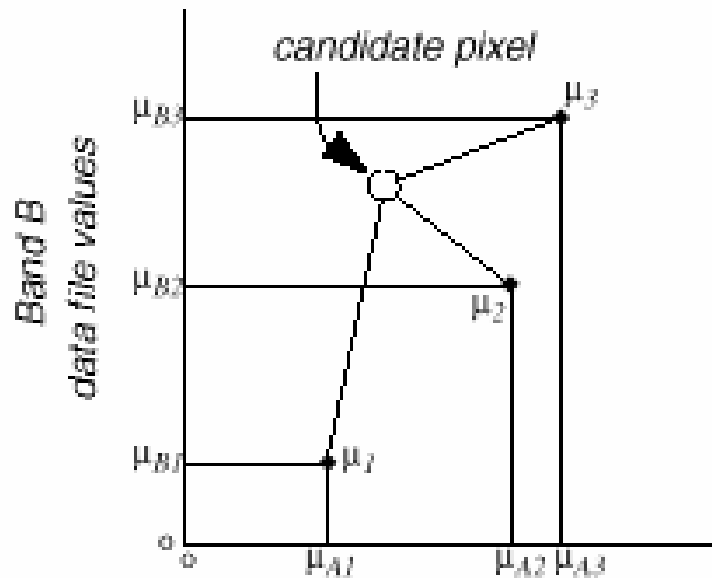
## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## LIMITACIONES DEL ALGORITMO DE MÍNIMA DISTANCIA



Sobreclasificación de píxeles

No considera la variabilidad de las clases

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

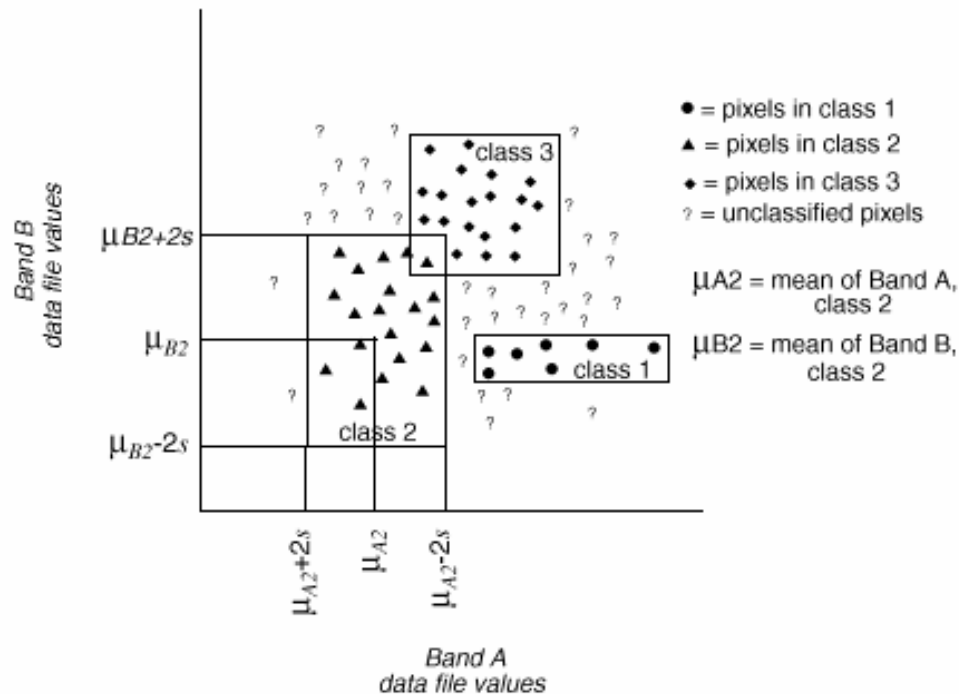
Clasificación con Mínima Distancia

Limitaciones del algoritmo  
Mínima Distancia





## LIMITACIONES DEL ALGORITMO DE PARALELEPÍPEDOS



Algunos píxeles quedan sin clasificar

A veces algunos píxeles pueden clasificarse erróneamente



Clasificación con Paralelepípedos

Limitaciones del algoritmo Paralelepípedos

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## ALGORITMOS CLÁSICOS CONTEXTUALES

- 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.

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

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

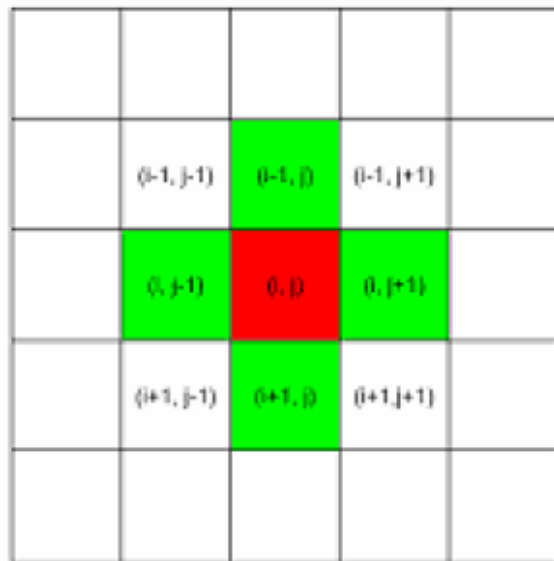
- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

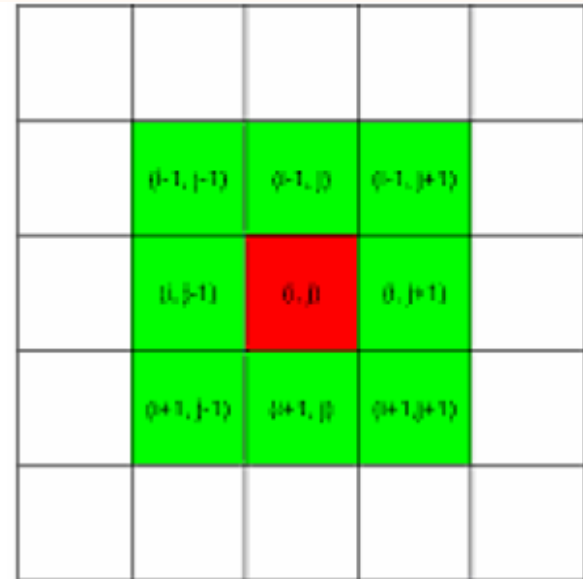
- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## DEFINICION 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.

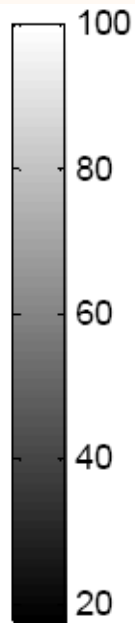
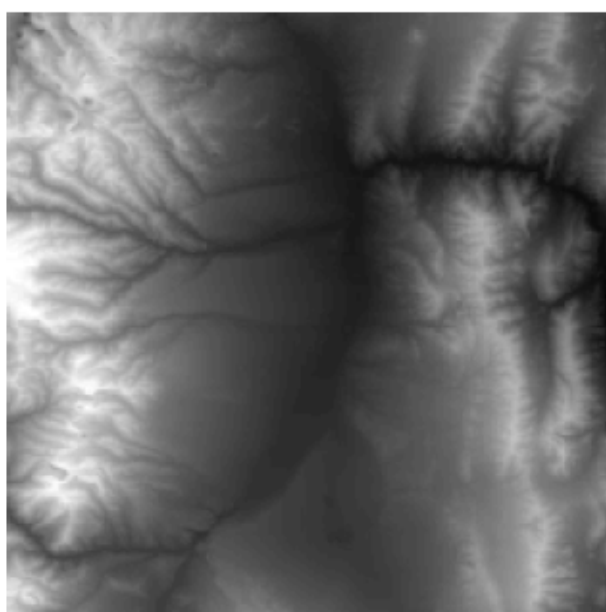


*Vecindad de Neumann*

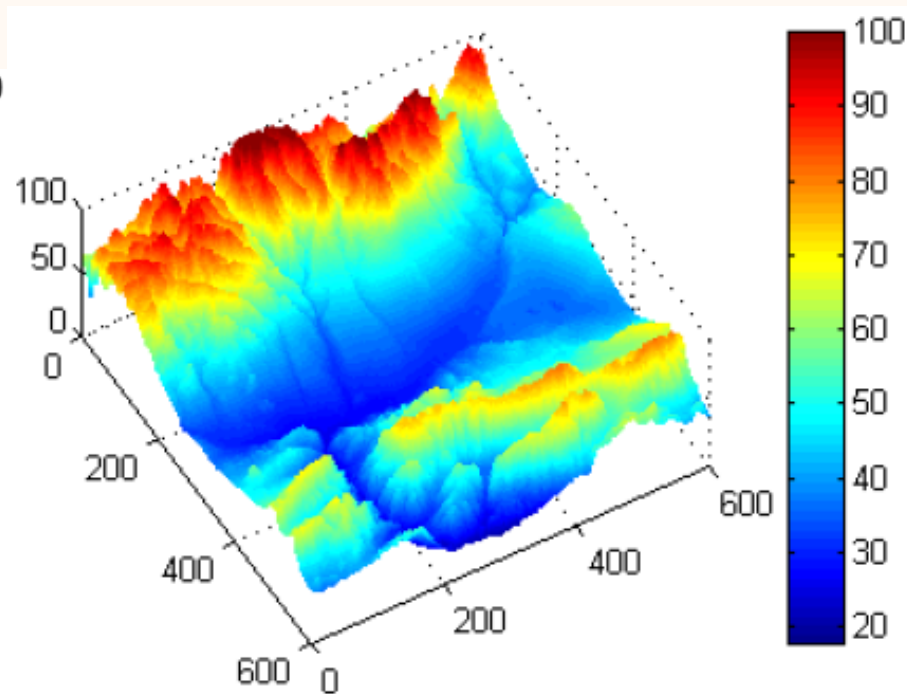


*Vecindad de Moore*

## SIMULACIÓN DEL PROCESO DE DERRETIMIENTO DE NIEVE



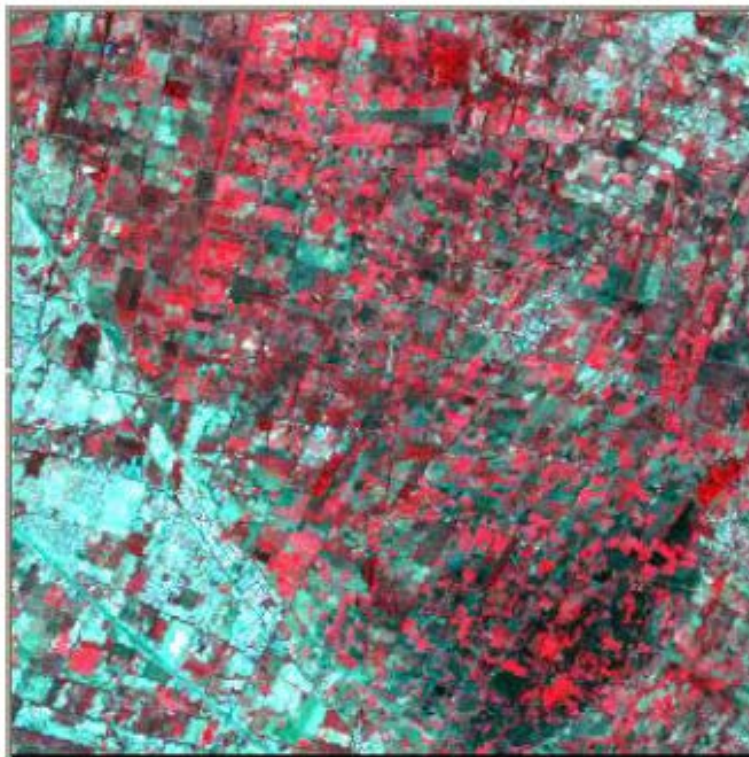
*DEM (Digital Elavation Model)*



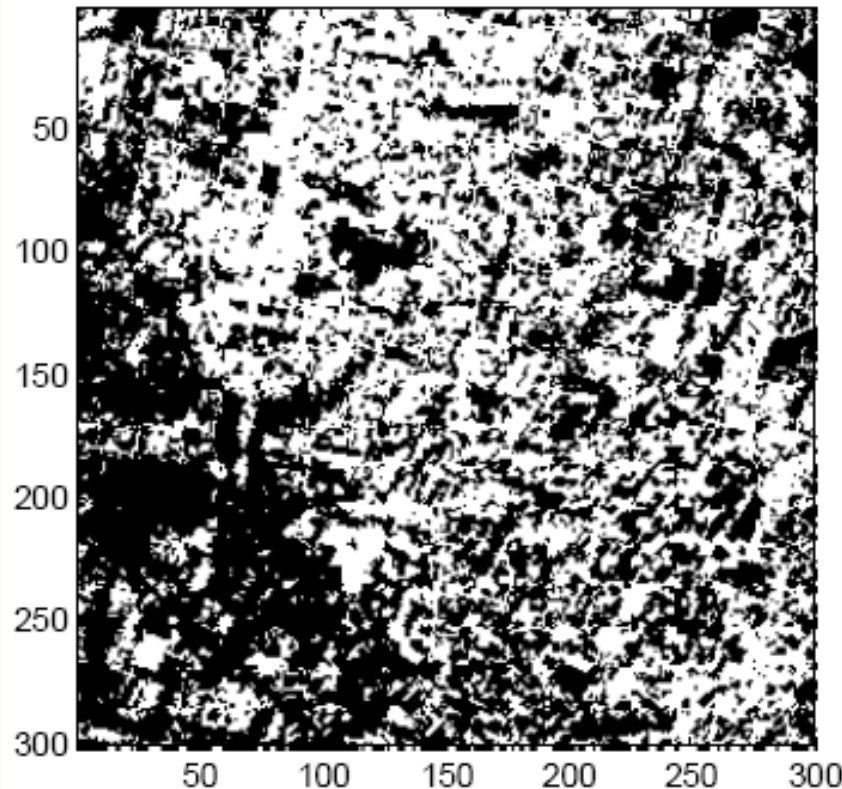
*DEM 3D con MATLAB*



## SIMULACIÓN DE DAÑOS PRODUCIDOS EN VEGETACIÓN POR INUNDACIÓN

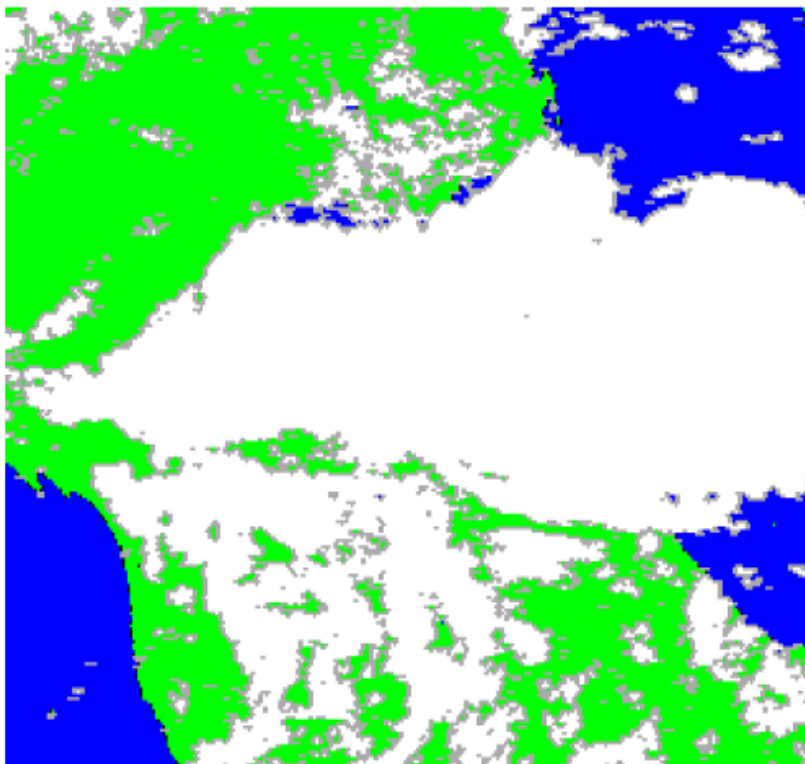


*Imagen original*

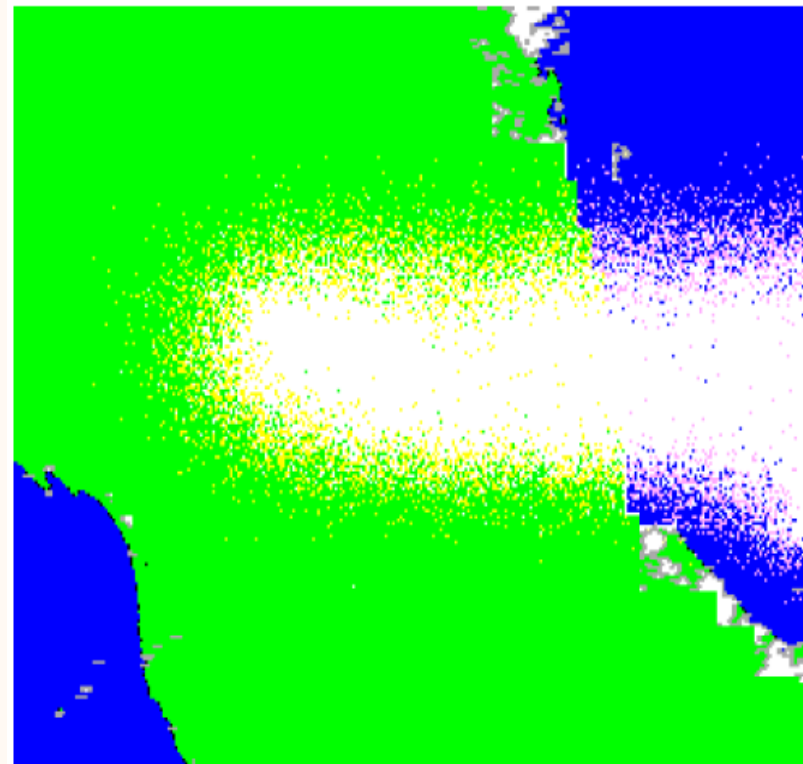


*Imagen clasificada*

## SIMULACIÓN DE NUBES EN IMÁGENES DE SATÉLITE

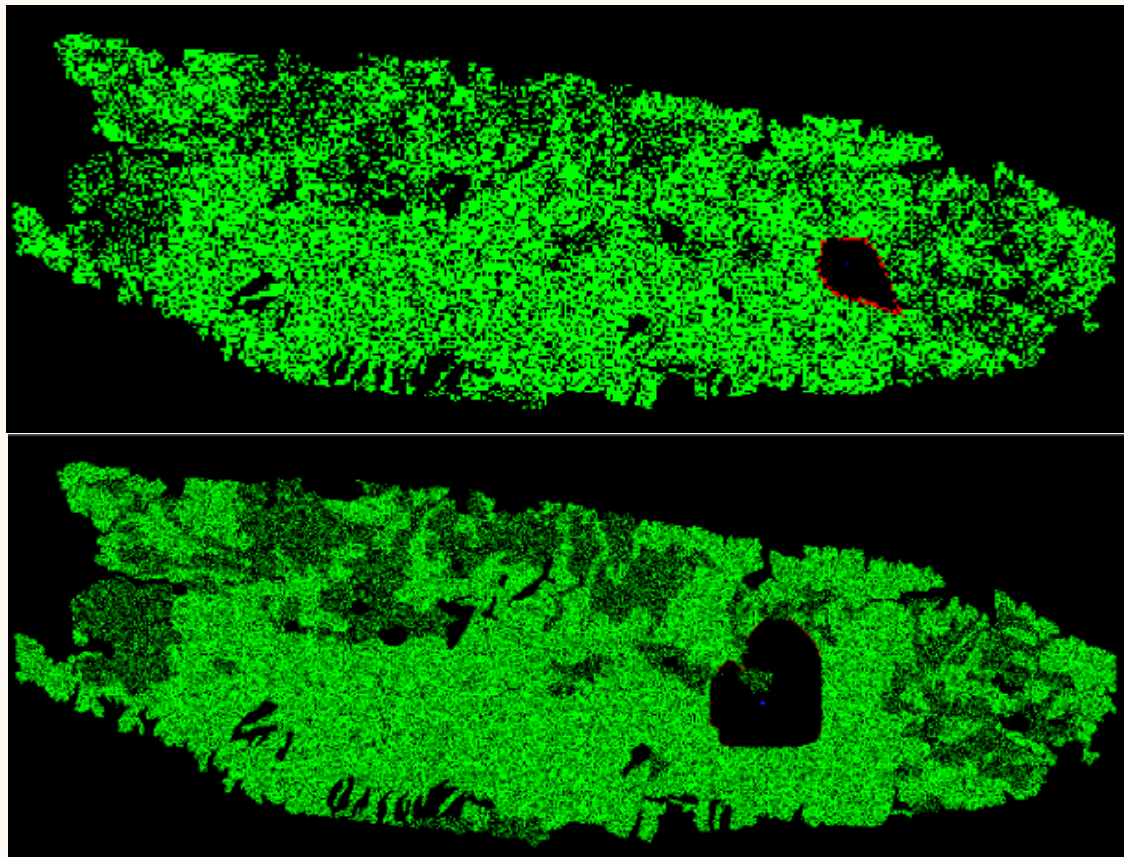


*Imagen NOAA14 actual. Clasificación (tierra, mar, nube y borde de nube).*



*Imagen simulada con AC. 3 fuentes de partículas.*

## SIMULACIÓN DE PROPAGACIÓN DE INCENDIOS



*Zona ardiendo y zona quemada*

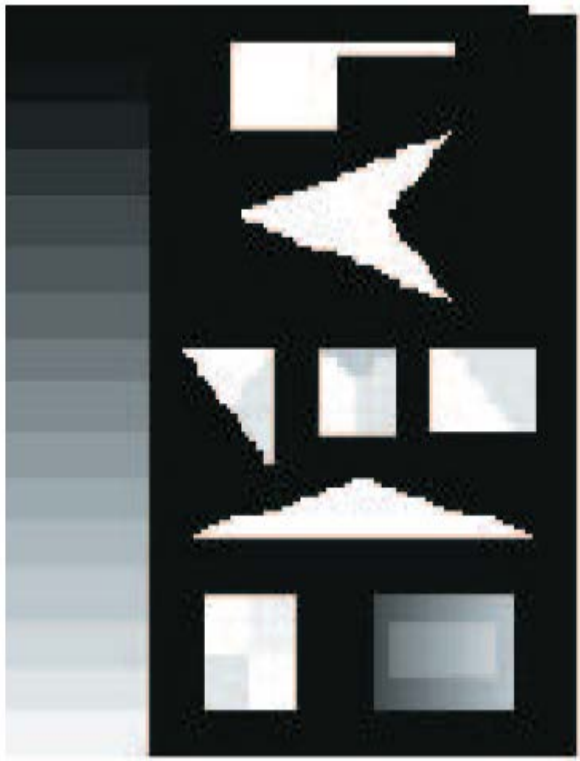


## PROCESAMIENTO DE IMÁGENES

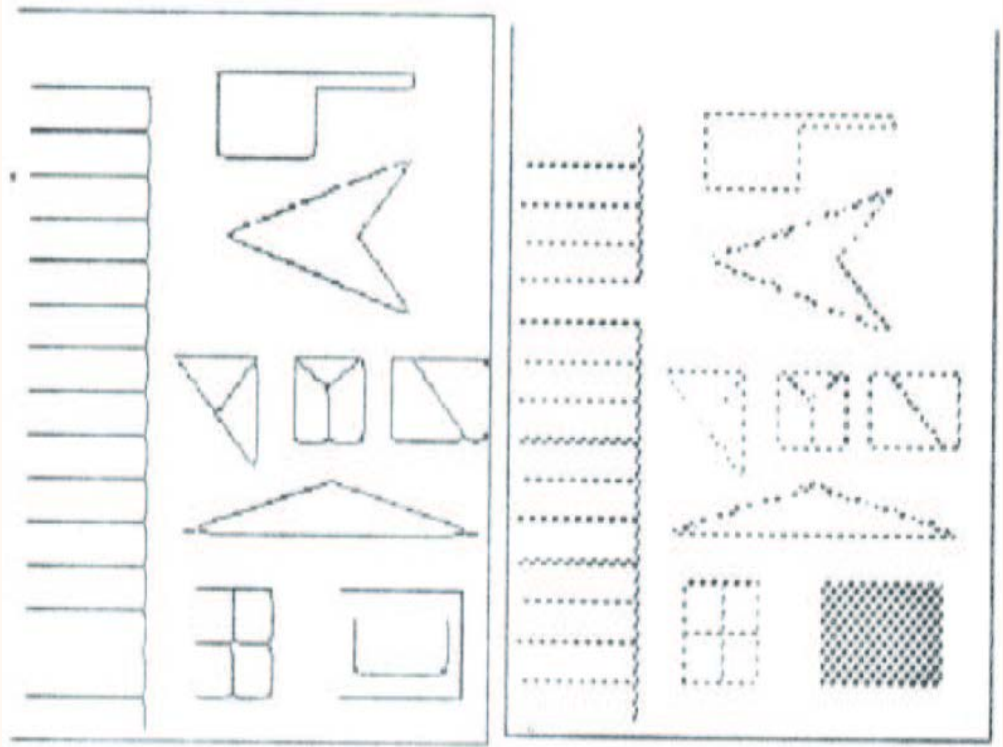


(a) Imagen original, (b) Imagen con ruido, (c) Filtro Gaussiano, (d) Filtro ACs

## DETECCIÓN DE BORDES Y FRONTERAS

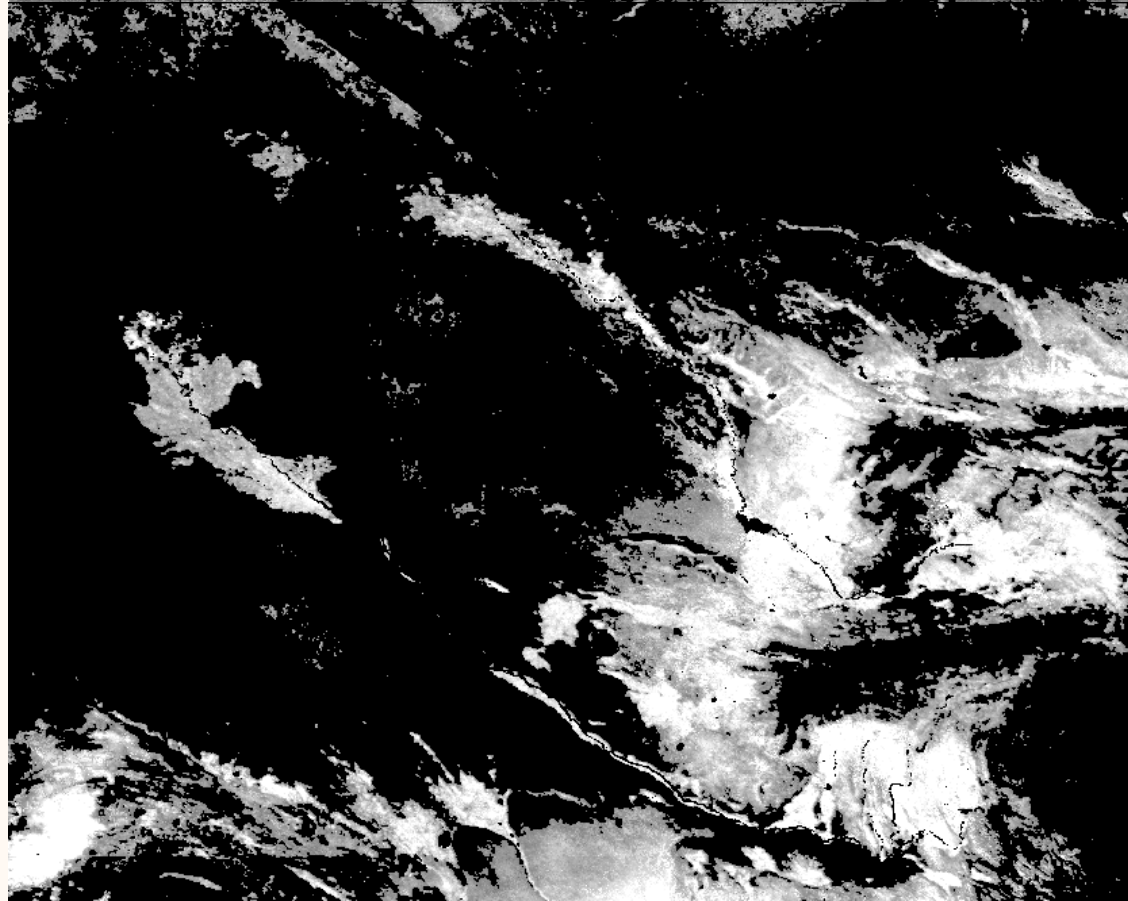


*Imagen original*



*(a) Algoritmo SUSAN, (b) Algoritmos ACs*

## SIMULACION CON AUTOMATAS CELULARES



*Simulación sencilla con autómatas celulares*

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**



## PÍXELES CAÓTICOS

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	7	8	8	8	8	8	8	7	5	4	4	5	5
19	7	3	7	7	7	7	7	8	8	8	8	9	8	7	5	4	4	5	5
20	7	5	7	7	7	7	7	8	8	8	8	9	9	7	5	4	4	5	5
21	7	5	5	7	7	7	8	8	8	8	8	9	7	5	4	4	7	8	8
22	7	7	5	7	7	7	7	8	8	8	8	8	8	7	7	7	7	8	8

**Clasificación ISODATA**

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	7	8	8	8	8	8	8	7	5	4	4	5	5
19	7	3	7	7	7	7	7	8	8	8	8	9	8	7	5	4	4	5	5
20	7	5	7	7	7	7	7	8	8	8	8	9	9	7	5	4	4	5	5
21	7	5	5	7	7	7	8	8	8	8	8	9	7	5	4	4	7	8	8
22	7	7	5	7	7	7	7	8	8	8	8	8	8	7	7	7	7	8	8

**Clasificación Mínima Distancia**

## PÍXELES CAÓTICOS

[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)	

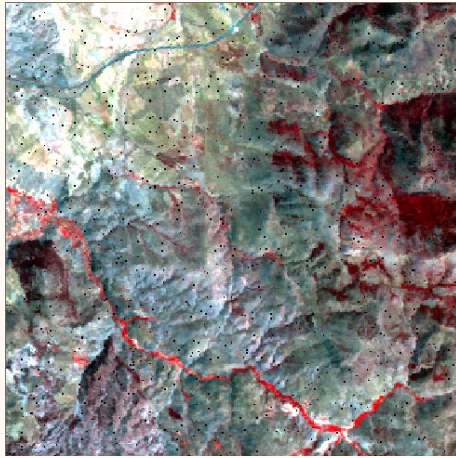
Row	158	159	160	161	162	163	164	165	166	167	168	169	170
0	34	70	46	35	35	33	46	54	46	58	67	66	63
1	34	61	51	44	44	47	53	45	47	66	67	67	67
2	43	40	55	52	52	56	47	44	54	66	65	74	69
3	65	60	49	54	54	54	46	52	64	63	70	70	70
4	73	68	62	51	49	41	61	72	64	59	67	68	71
5	73	67	47	45	50	56	62	64	63	63	74	74	75
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**

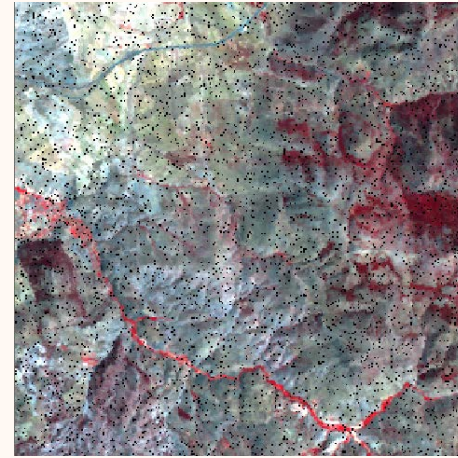
**Archivo SIG**

**Capas de la imagen original**

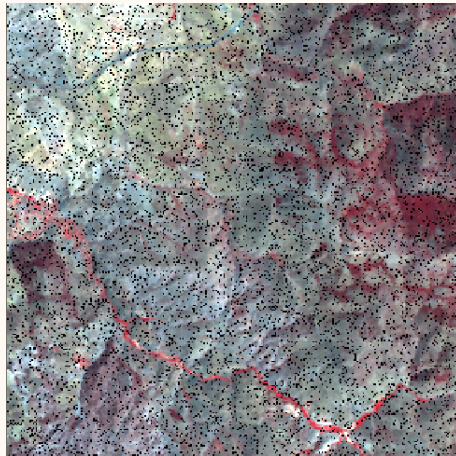
## PÍXELES DE RUIDO IMPULSIVO



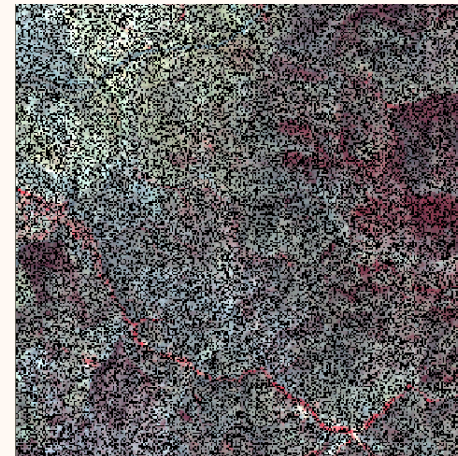
**1% ruido**



**5% ruido**



**10% ruido**

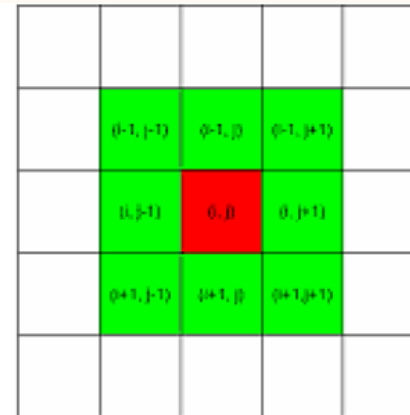
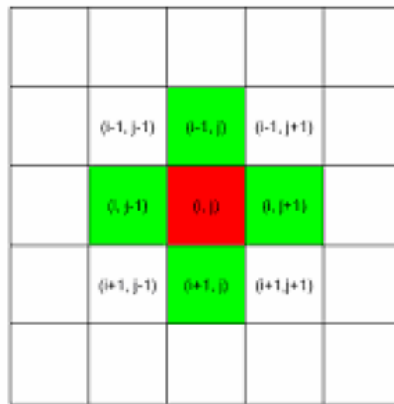


**50% ruido**

## OBJETIVOS DEL ALGORITMO ACA

### OBJETIVOS DE ACA

1. Eliminamos el ruido: clasificamos bien los píxeles con ruido impulsivo.
2. Afinamos la clasificación en píxeles frontera: píxeles dudosos de frontera (si está próximo a dos clases, tomar la clase mayoritaria de alrededor).



## MAJORITY FILTER

### MAJORITY FILTER

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

If ( $n_i > n_j$  &&  $n_i > n_t$  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 DEL AUTÓMATA CELULAR

### 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 8 pixeles vecinos.
- La función de transición ( $f$ ) debe clasificar correctamente cada pixel de la imagen, usando una fusión de clasificación espectral-contextual.

## DEFINICIÓN DEL AUTÓMATA CELULAR

# **ALGORITMO ACA (VERSIÓN 1) - SECUENCIAL -**



## DEFINICIÓN DE LOS ESTADOS DEL AUTÓMATA CELULAR

### ESTADOS DEL AUTÓMATA CELULAR

- Clase: clases del .sig.

### VALORES DE ENTRADA DEL AUTÓMATA CELULAR

Clases de la vecindad, clasificación espectral del píxel actual

, donde 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).



## VERSION 1 DEL ALGORITMO ACA

### ACA CON 3 ITERACIONES (SECUENCIAL)

---

- *Primera iteración*: clasifica píxeles **buenos** (aquellos que no son caóticos ni ruidosos) → espectral.
  - *Segunda iteración*: clasifica píxeles **caóticos** o frontera usando Majority Filter entre clases dudosas → contextual.
  - *Tercera iteración*: clasifica píxeles **ruidosos** usando Majority Filter entre las clases vecinas → contextual.
-

## VENTAJAS DEL ALGORITMO ACA (VERSIÓN 1)

### VENTAJAS

- Gran mejora en la clasificación de los píxeles de la imagen, tanto caóticos como ruidosos.
- Mezcla entre clasificación espectral-contextual (más información).
- Clasificación incremental (cada iteración trabaja sobre la anterior, donde los píxeles clasificados son más seguros que en la actual).
- Se pueden usar algoritmos clásicos: mínima distancia, paralelepípedos, mahalanobis, etc.

## DEFINICIÓN DEL AUTÓMATA CELULAR

# **ALGORITMO ACA (VERSIÓN 2) - RECURSIVO -**

## DEFINICIÓN DE LOS ESTADOS DEL AUTÓMATA CELULAR

### 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

### 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

### 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:

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

## VERSION 2 DEL ALGORITMO ACA (FUNCIÓN PRINCIPAL)

### *Supervised Spectral-Contextual Classification Algorithm Based On Cellular Automata (SSCABOCA)*

```

1 function ACA (numIterations, threshold)
2 //
3 //
4 //
5 for i:=0 to numIterations do
6     for x:=0 to tamX do
7         for y:=0 to tamY do
8             if pixelx,y.classified ≠ true then
9                 res1:=spectralACA(x,y);
10                if res1≠∅ then
11                    res2:=contextualACA(res1);
12                endif
13            endif
14        endfor
15    endfor
16    threshold:=threshold+incremental;
17 endfor
18endfunction

```

**Table 1.** ACA Supervised Classification (General Algorithm)

## VERSION 2 DEL ALGORITMO ACA (FUNCIÓN ESPECTRAL)

```
1 function spectralACA (numClasses, signatures, )
2 //
3 //
4 //
5 res1:=∅;
6 for i:=0 to numClasses do
7     if pixelx,y.distancesi ≤ sthreshold then
8         res1:=res1 ∪ classi;
9     endif
10 endfor
11 endfunction
```

**Table 2.** ACA Supervised Classification (Spectral Subalgorithm)



## VERSION 2 DEL ALGORITMO ACA (FUNCIÓN CONTEXTUAL)

```
1 function contextualACA (res1)
2 //
3 //
4 //
5 if size(res1)=1 then
6     if (res1)~=noiseClass then
7         pixelxy.class:=(res1);
8         pixelxy.quality:=numIteration;
9         if neighbourhoodClassesType(pixelxy)=1 then
10             pixelxy.type:=focusPixel;
11         else
12             pixelxy.type:=edgePixel;
13         endif
14     elseif (res1)~=noiseClass then
15         pixelxy.class:=bayesNeighbourhood ();
16         pixelxy.quality:=numIteration;
17         pixelxy.type:=noisePixel;
18     endif
19 else
20     pixelxy.class:=bayesNeighbourhoodClass ();
21     pixelxy.quality:=numIteration;
22     pixelxy.type:=caoticPixel;
23 endif
24 endfunction
```

**Table 3.** ACA Supervised Classification (Contextual Subalgorithm)

## VENTAJAS DEL ALGORITMO ACA (VERSIÓN 2)

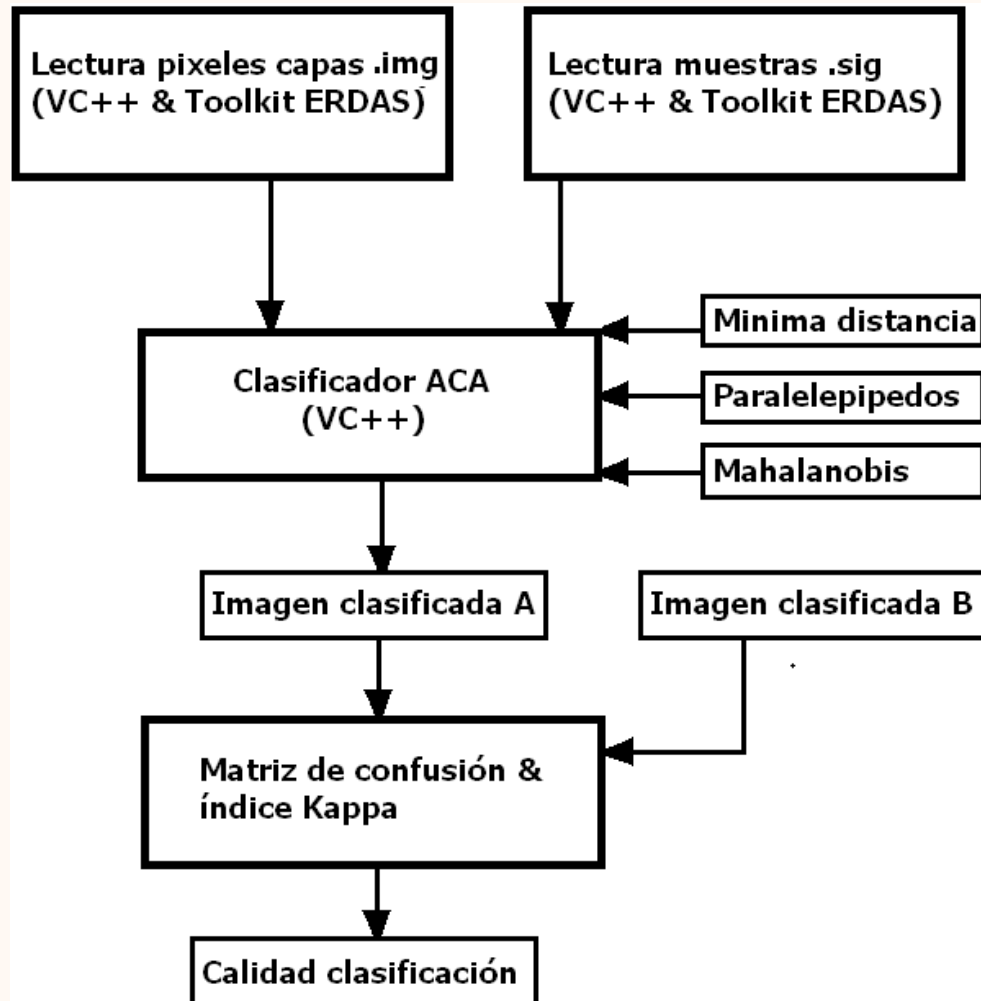
### VENTAJAS

- Todas las ventajas de la primera versión ACA, más:
- Clasificación “jerárquica” de los píxeles de la imagen.
- Detección del tipo de pixel: bueno, frontera, caótico, ruidoso.

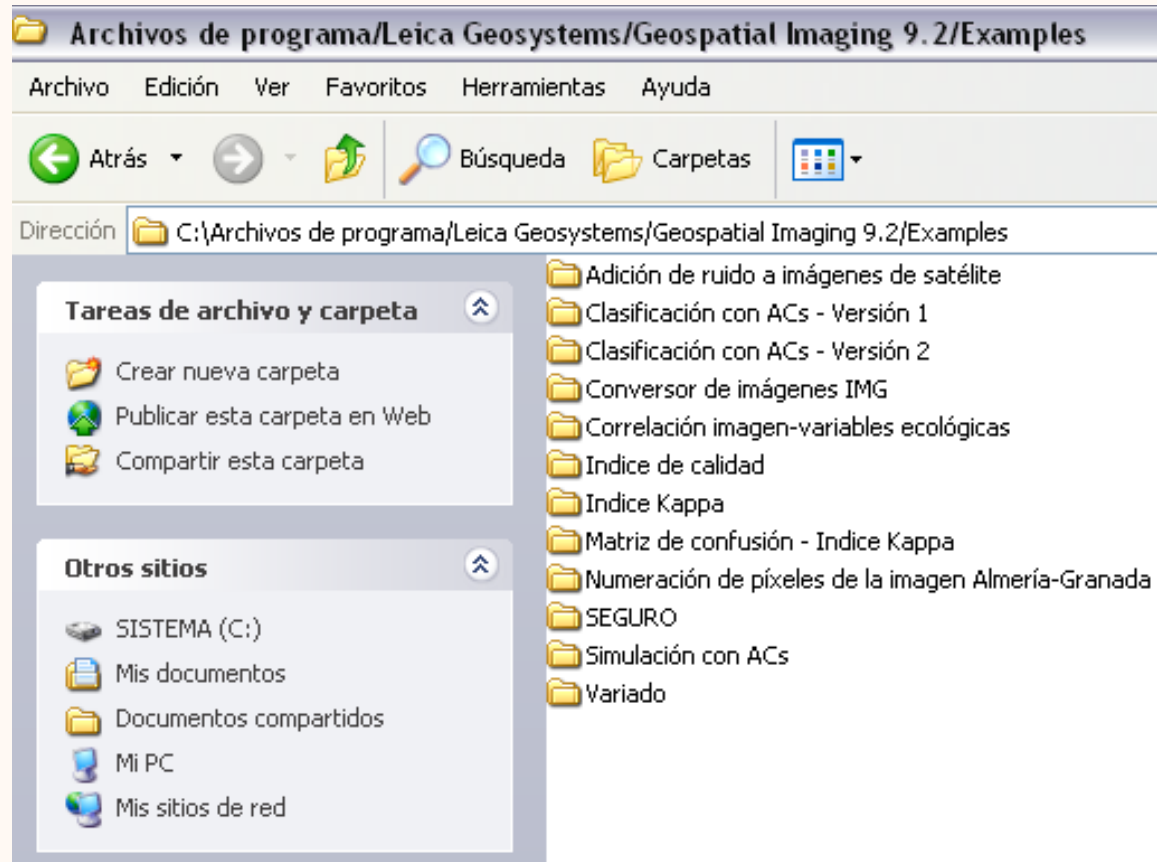
### DESVENTAJA

- Incremento del coste computacional.

## ARQUITECTURA ACA

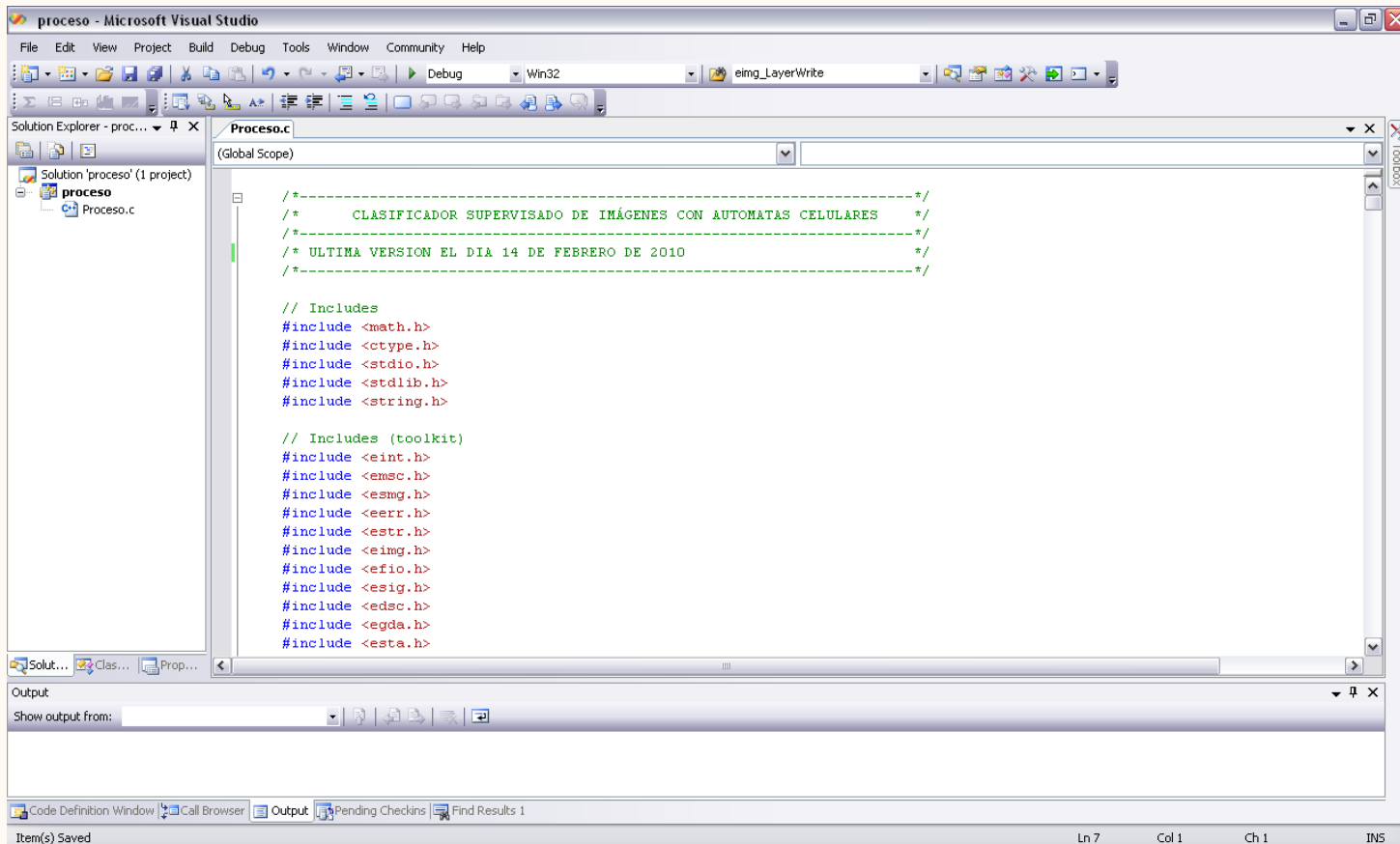


## PROGRAMAS IMPLEMENTADOS



[Programas implementados en Toolkit ERDAS 9.1](#)

## CÓDIGO FUENTE



The screenshot shows the Microsoft Visual Studio IDE with the following details:

- Window Title:** proceso - Microsoft Visual Studio
- Menu Bar:** File, Edit, View, Project, Build, Debug, Tools, Window, Community, Help
- Toolbar:** Standard Visual Studio toolbar with icons for File, Edit, View, Project, Build, Debug, Tools, Window, Community, Help.
- Solution Explorer:** Shows a project named 'proceso' with a file named 'Proceso.c'.
- Code Editor:** Displays the source code for 'Proceso.c' with the following content:

```
/*-----*/
/*  CLASIFICADOR SUPERVISADO DE IMÁGENES CON AUTOMATAS CELULARES  */
/*-----*/
/*  ULTIMA VERSION EL DIA 14 DE FEBRERO DE 2010  */
/*-----*/

// Includes
#include <math.h>
#include <ctype.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

// Includes (toolkit)
#include <eint.h>
#include <emsc.h>
#include <esmg.h>
#include <eerr.h>
#include <estr.h>
#include <eimg.h>
#include <efio.h>
#include <esig.h>
#include <edsc.h>
#include <egda.h>
#include <esta.h>
```
- Output Window:** Empty, with a dropdown menu for 'Show output from:'.
- Status Bar:** Shows 'Item(s) Saved', 'Ln 7', 'Col 1', 'Ch 1', and 'IMS'.

[Ver código fuente ACA](#)

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

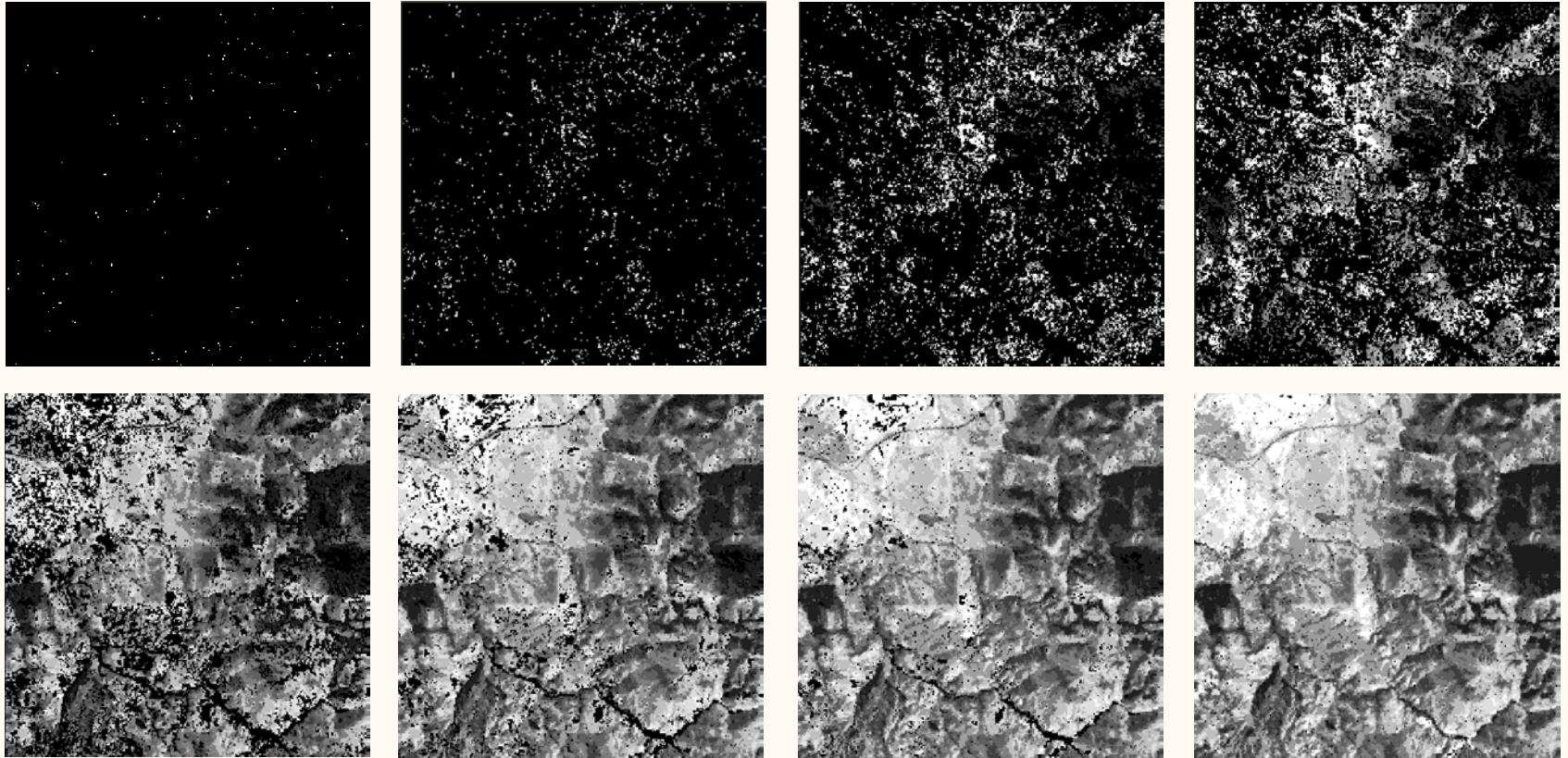
- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACIÓN RECURSIVA



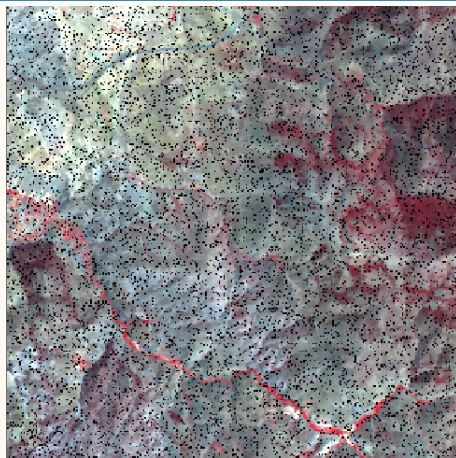


## CLASIFICACIÓN JERARQUIZADA EN NIVELES DE CALIDAD

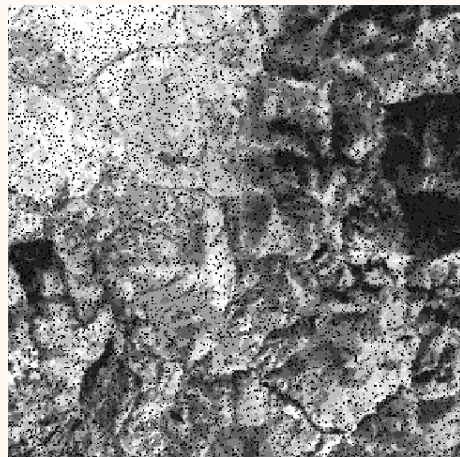


**Clasificación jerarquizada en 8 iteraciones del AC**

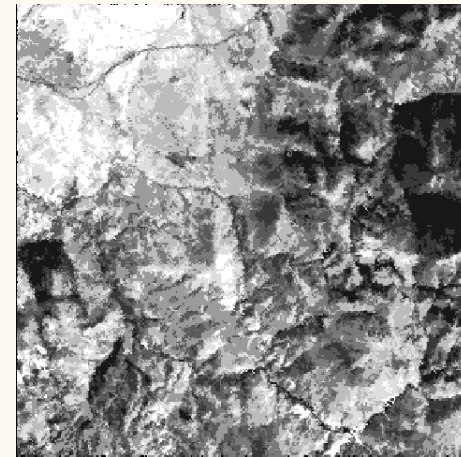
## CLASIFICADOR DE MÍNIMA DISTANCIA VS CLASIFICADOR ACA



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



**Clasificación con Mínima Distancia**

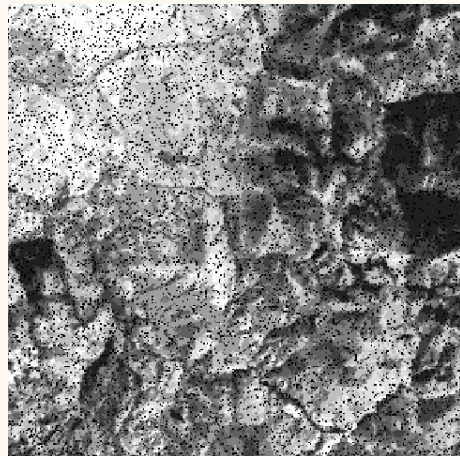


**Clasificación con ACA**

## MATRIZ DE CONFUSIÓN DEL ALGORITMO DE MÍNIMA DISTANCIA



**Clasificación inicial con ISODATA**



**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	<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>

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

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



## MATRIZ DE CONFUSIÓN DEL ALGORITMO ACA



**Clasificación inicial con ISODATA**



**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	<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>

**Matriz de confusión del algoritmo ACA**  
**Arriba: ISODATA. 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**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## FUTURAS MEJORAS AL ALGORITMO ACA

### TRABAJOS FUTUROS

- Utilizar agentes software para reducir el coste computacional.
- Integrar todas las herramientas en un solo software.
- Nuevas versiones de ACA.



## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## CLASIFICACION DE IMÁGENES DE SATÉLITE MEDIANTE ACs

- **Introducción**
- **Algoritmos de clasificación espectrales**
- **Algoritmos de clasificación contextuales**
- **Autómatas celulares usados en simulación**
- **Clasificación de imágenes con autómatas celulares**
- **Resultados obtenidos**
- **Trabajos futuros**
- **Preguntas**

## PREGUNTAS SOBRE AUTÓMATAS CELULARES

- ¿Están bien definidos los ACs?
- ¿Qué utilidades puede tener la clasificación jerarquizada?
- ¿Es buena idea darle al ACA los valores min.dist. y que, a través de reglas, elija si usar vecindad?
- ¿Es obligatorio usar siempre vecindad en ACs?
- ¿Se puede usar en las reglas de ACs datos distintos de los estados de los vecinos (ej: clasificación espectral)?

## PREGUNTAS SOBRE RUIDO IMPULSIVO Y CALIDAD DE RESULTADOS

- ¿Es común el ruido impulsivo? ¿Hasta qué porcentaje puede surgir?
- En general, ¿son comunes los píxeles caóticos o frontera (en imágenes homogéneas y heterogéneas)?
- ¿En qué otro tipo de casos se podría aplicar ACA (ej: nubes)?
- ¿Qué método recomienda para comprobar la calidad de la clasificación?
- ¿Cómo podemos obtener imágenes clasificadas por expertos (trabajo de campo)?

## PREGUNTAS SOBRE MATRIZ DE CONFUSIÓN

- ¿Cómo aumenta el Threshold en cada iteración del AC?
- ¿Cómo podríamos mejorar la originalidad del algoritmo?
- ¿Podríamos usar la Tesis de Rosa Ayala?

### CLASIFICACION SUPERVISADA DIFUSA

#### INDICES DE CALIDAD PARA TODAS LAS COMBINACIONES

Iteración (no supervisada)	Iteración (supervisada)	Parámetros (no supervisada)	Parámetros (supervisada)	Separabilidad	Calidad	Threshold	Indice de calidad
1	1	Converge: 0.0 Maxiter: 6 Initmean: calc Num clases: 5	Non-parametric Rule: Non Parametric Rule: Max Overlap Options: Par Unclass Opciones: Par	1.345949	0.977267	0.544115	0.856268
1	2	Converge: 0.0 Maxiter: 6 Initmean: calc Num clases: 5	Non-parametric Rule: Non Parametric Rule: Min Overlap Options: Par Unclass Opciones: Par	1.345949	0.956672	0.459107	0.829863
1	3	Converge: 0.0 Maxiter: 6 Initmean: calc Num clases: 5	Non-parametric Rule: Non Parametric Rule: Mah Overlap Options: Par Unclass Opciones: Par	1.345949	0.893992	0.582127	0.844950
1	4	Converge: 0.0 Maxiter: 6 Initmean: calc Num clases: 5	Non-parametric Rule: Ppd Parametric Rule: Max Overlap Options: Par Unclass Opciones: Par	1.345949	0.977536	0.723342	0.901148
1	5	Converge: 0.0 Maxiter: 6 Initmean: calc Num clases: 5	Non-parametric Rule: Ppd Parametric Rule: Mah Overlap Options: Par Unclass Opciones: Par	1.345949	0.894026	0.761352	0.889772

## AGRADECIMIENTOS

**¡MUCHAS GRACIAS  
POR LA ATENCION!**

