

IMÁGENES DE SATÉLITE Y AUTÓMATAS CELULARES: NUEVAS TÉCNICAS DE CLASIFICACIÓN SUPERVISADA

A. Moisés Espínola Pérez
- PROYECTO SOLERES -

20 de febrero de 2012

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

ALGORITMOS DE CLASIFICACIÓN MÁS COMUNES

**Clasificación
no supervisada**

- **K-medias**

- **Isodata**

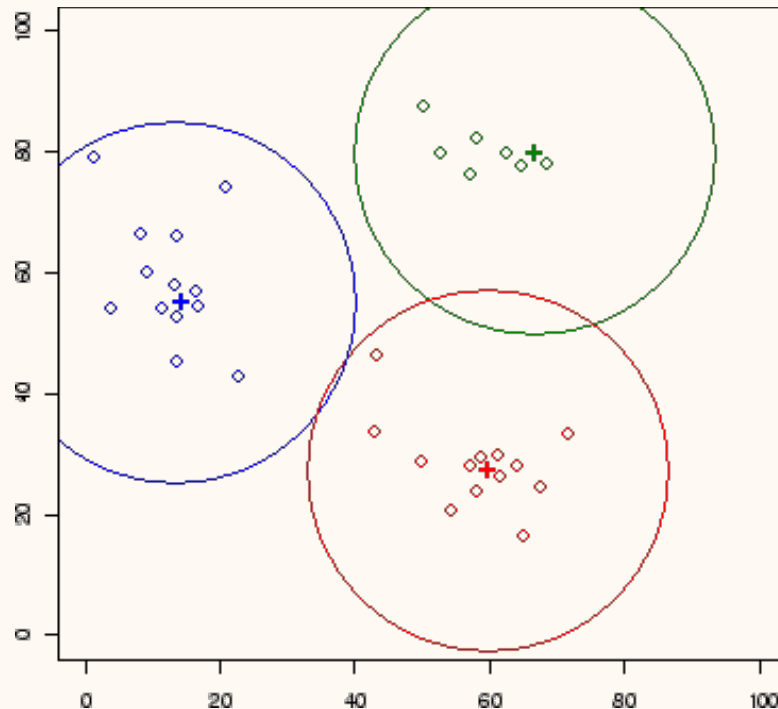
**Clasificación
supervisada**

- **Mínima distancia**

- **Paralelepípedos**

- **Máxima verosimilitud**

ALGORITMO DE MÍNIMA DISTANCIA



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

Clasificación con mínima distancia

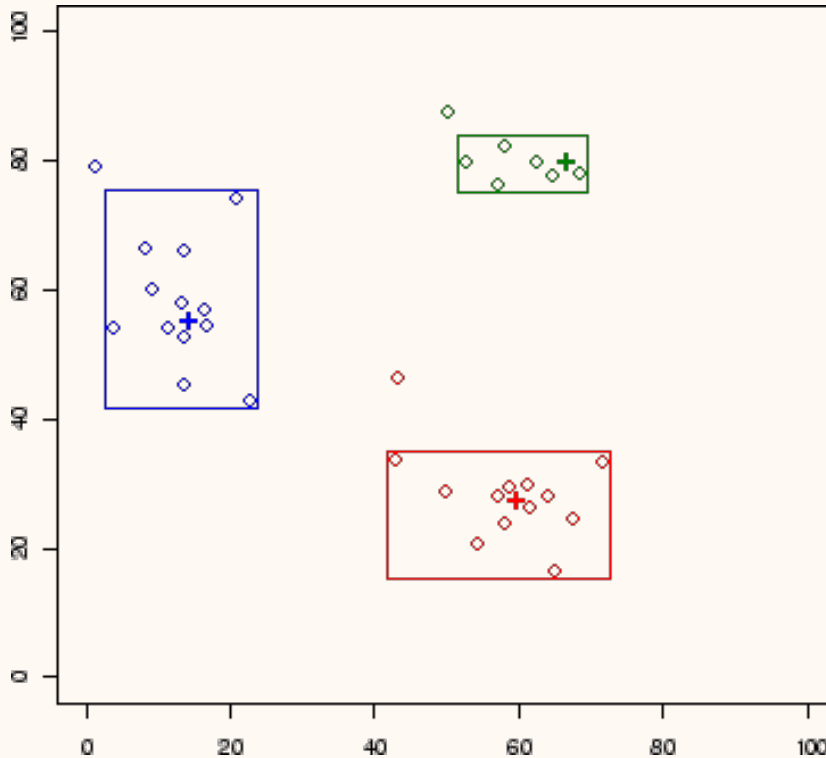
**Sobreclasificación de píxeles
(proclive a errores de comisión)**

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



INCONVENIENTES

ALGORITMO DE PARALELEPÍEDOS



Clasificación con paralelepípedos

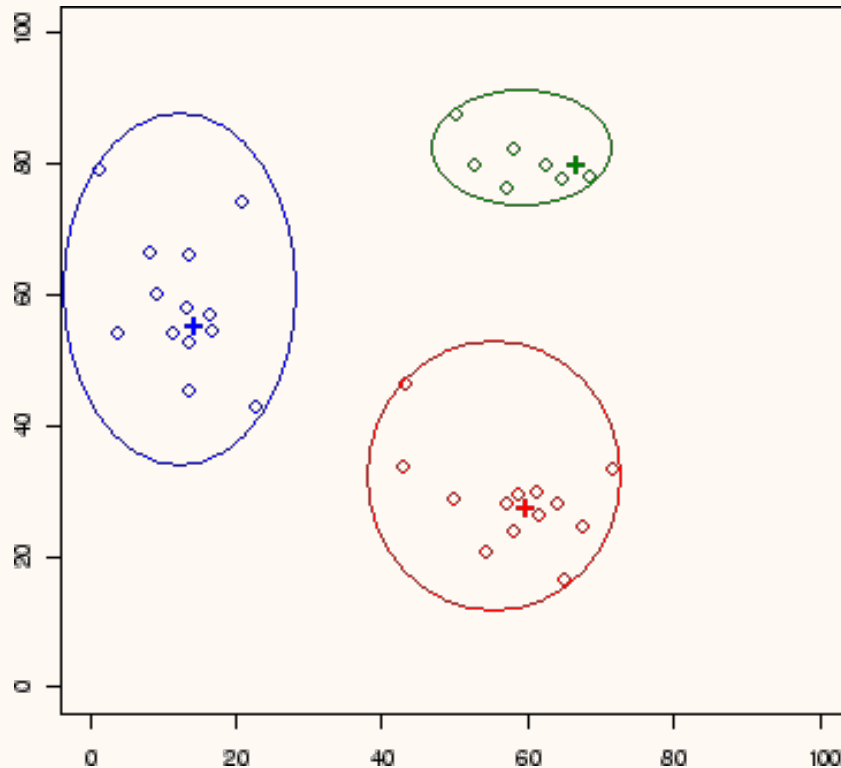
Algunos píxeles quedan sin clasificar

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

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

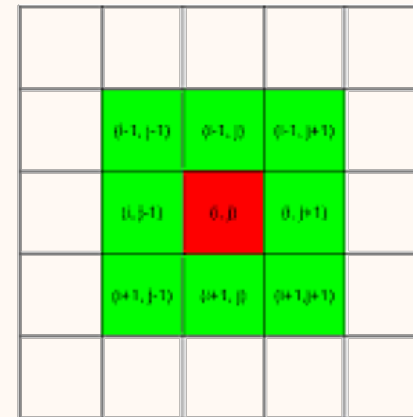
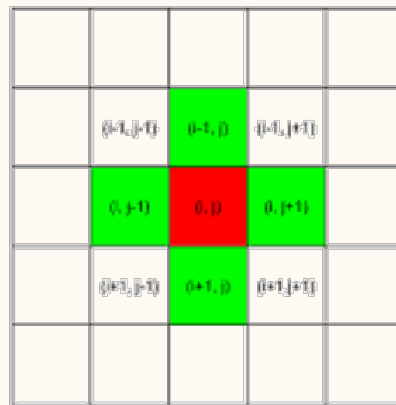
Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

OBJETIVOS DEL ALGORITMO ACA

OBJETIVOS DE ACA

1. Mejorar la calidad de la clasificación (VECINDAD).
2. Obtener una lista de píxeles inciertos, ruidosos, bordes (VECINDAD).
3. Clasificación jerarquizada 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.

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:

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

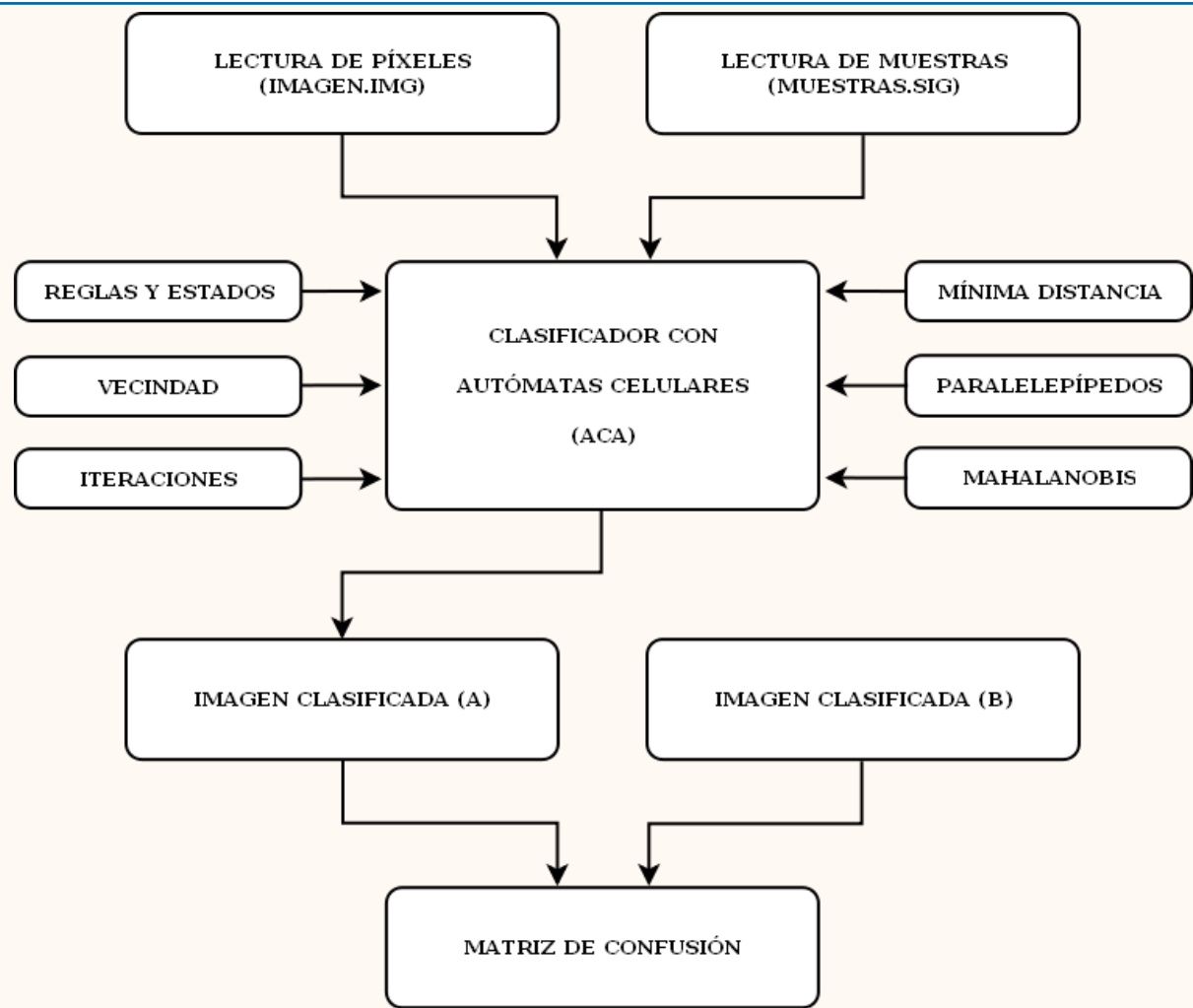
Si píxel caótico:

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

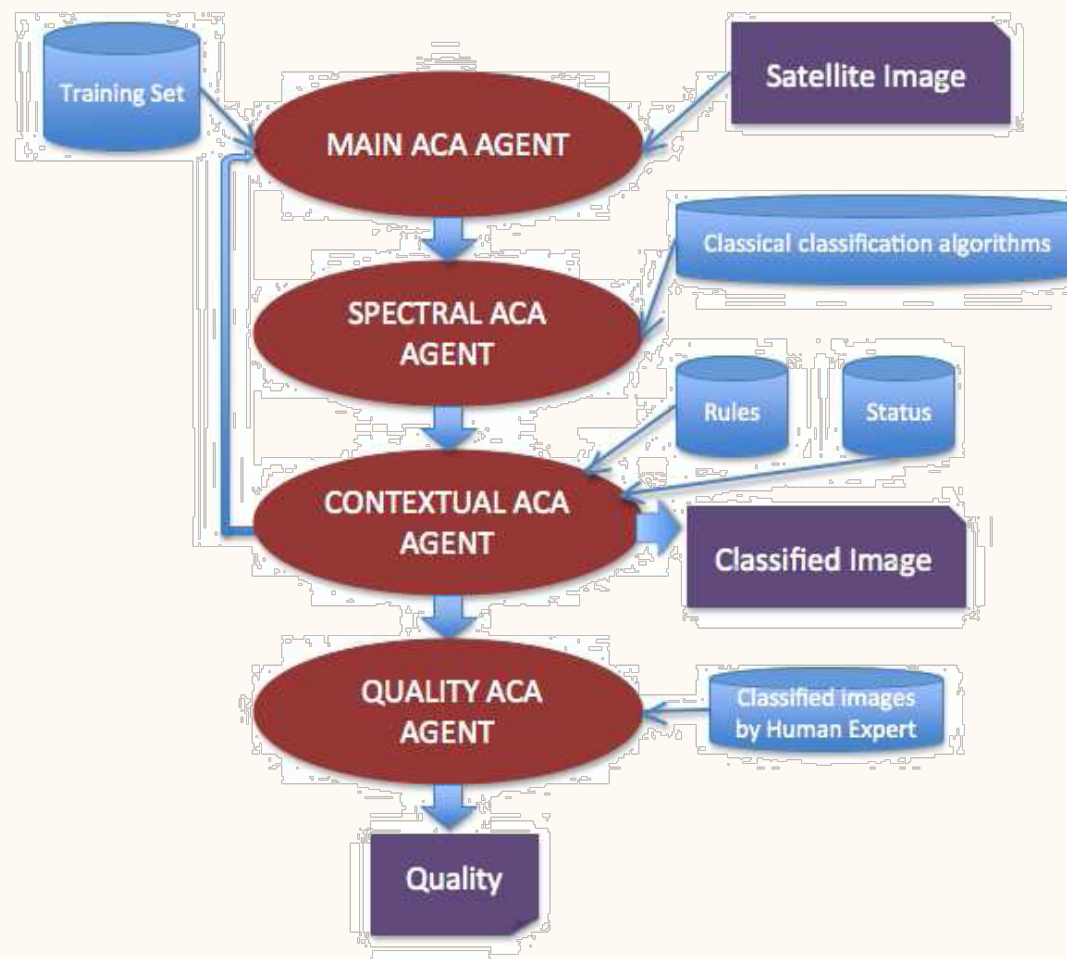
Si píxel ruidoso:

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

ARQUITECTURA COMPLETA DEL ALGORITMO ACA



ARQUITECTURA COMPLETA DEL ALGORITMO ACA (AGENTES)



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  $e_i.classified \neq true$  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)

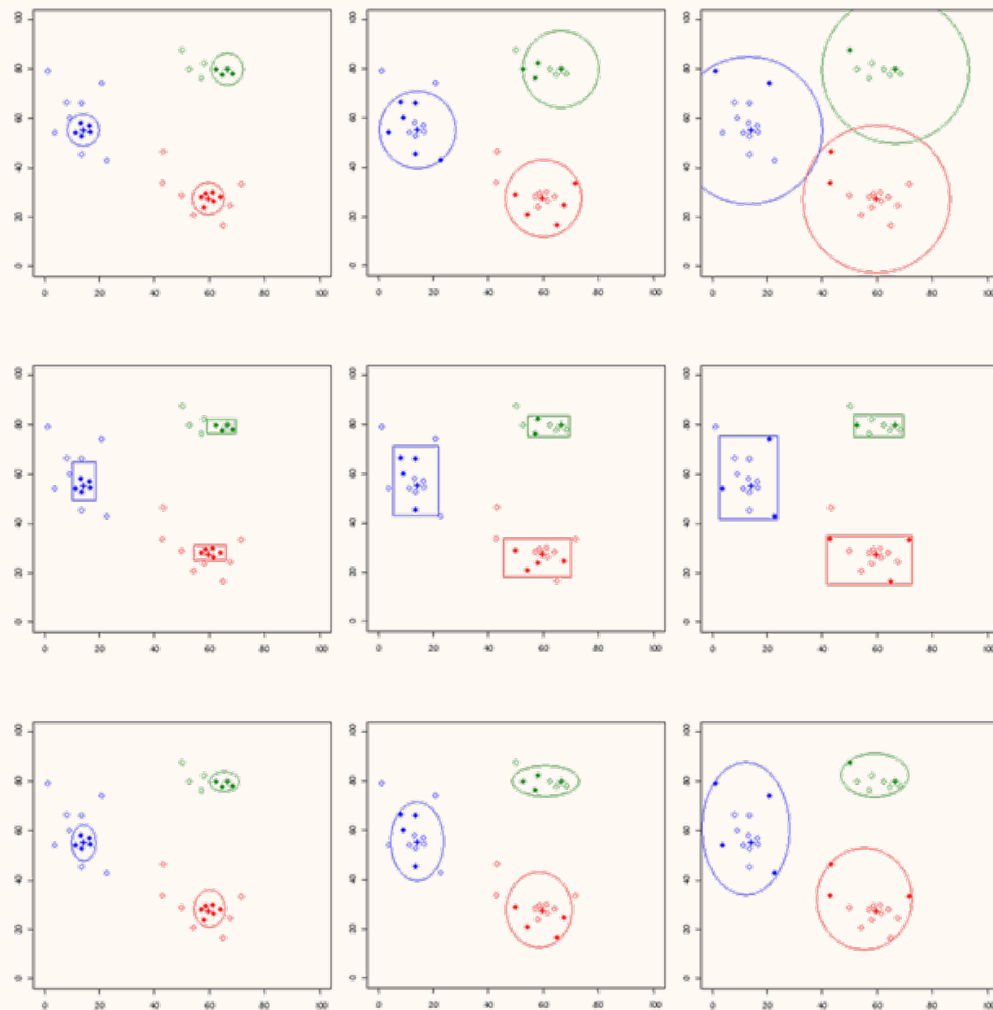
Entrada:
 $E = \{e_1, e_2, \dots, e_n\}$: conjunto de píxeles a clasificar

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       $e_i.type \leftarrow chaoticPixel$ ;
22  endif
    
```

SIMULACIÓN DEL PROCESO DE CLASIFICACIÓN CON ACA



Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

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

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

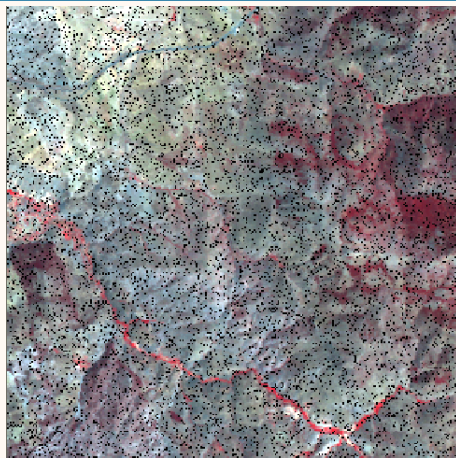
Matriz de confusión del algoritmo ACA

Píxeles bien clasificados:
72.974 / 90.601 = 80%

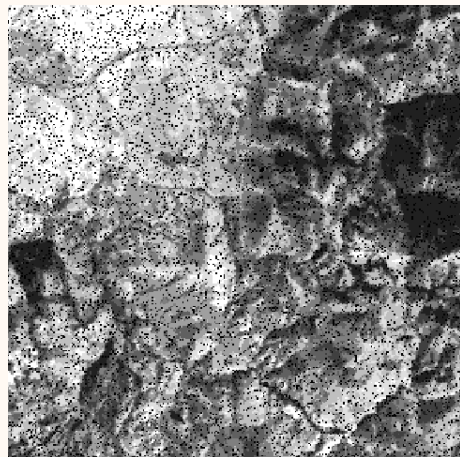
Píxeles bien clasificados:
75.899 / 90.601 = 84%

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

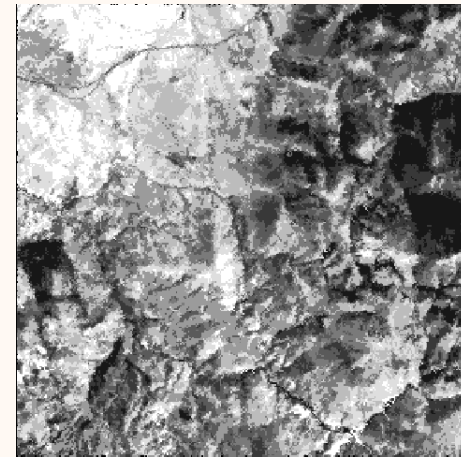
CLASIFICADOR DE MÍNIMA DISTANCIA VS CLASIFICADOR ACA



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

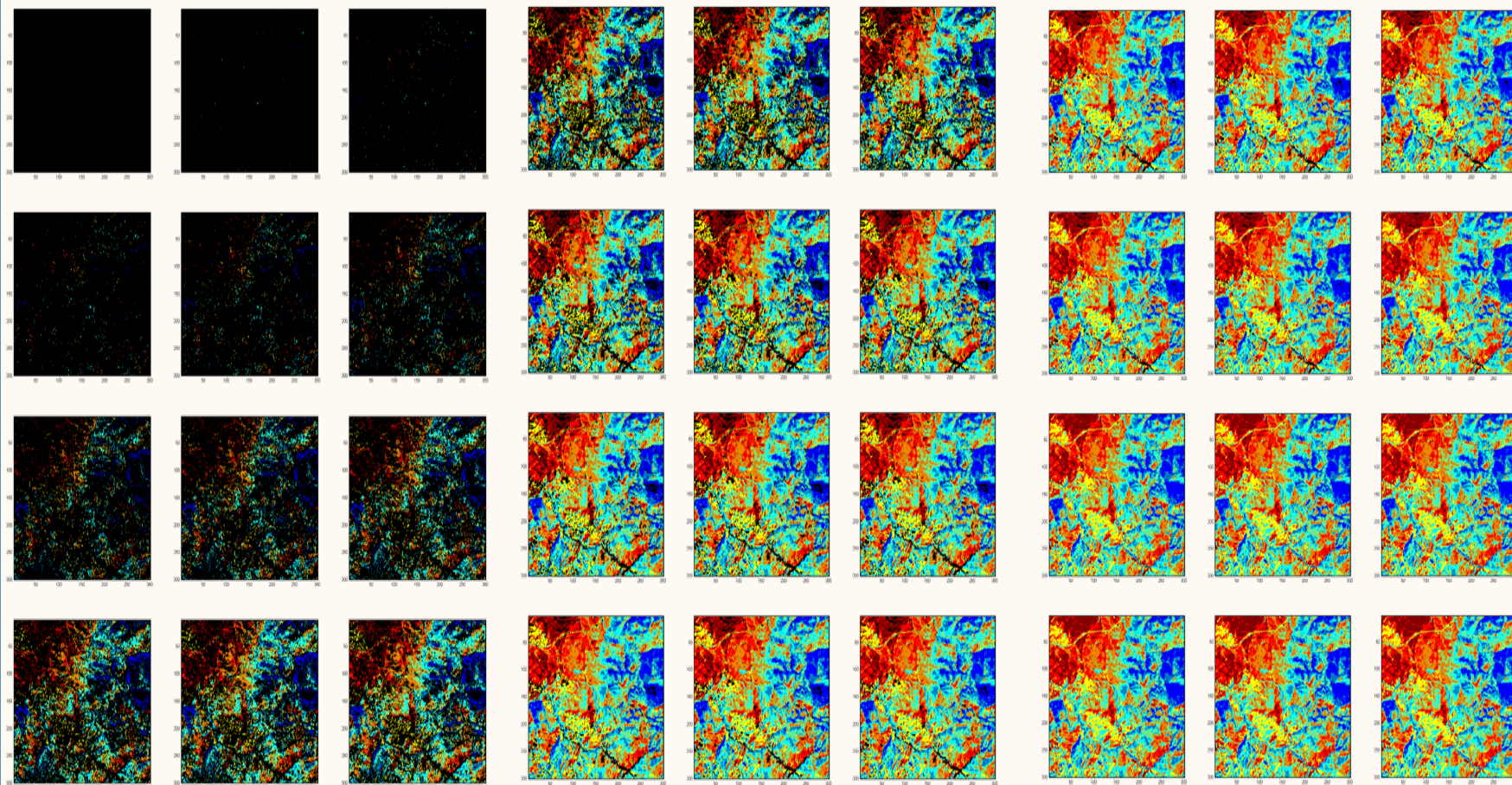


Clasificación con Mínima Distancia



Clasificación con ACA

CLASIFICACIÓN JERARQUIZADA EN 100 NIVELES DE FIABILIDAD



Clasificación jerarquizada en 100 iteraciones del AC

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

MATRIZ DE CONFUSIÓN – CÁLCULOS DERIVADOS

The performance measures that are deduced from the confusion matrix are calculated as follows.

- accuracy(*auc*):** the overall correctness of the model calculates as the sum of correct predictions divided by the total number of predictions, i.e. $auc = \frac{\sum tp_i}{\sum tp_i + \sum e_{jk}}$ where tp_i are all elements falling along the diagonal of the confusion matrix, and e_{jk} are all elements falling off the diagonal ($i, j, k \in \{A, B, C\} \wedge j \neq k$).
- precision(*pr*):** a measure of the accuracy provided that a specific class has been predicted. It is defined by $pr_i = \frac{tp_i}{tp_i + \sum e_{ji}}$, where $i \in \{A, B, C\}$, and $j \in \{c | c \in \{A, B, C\} \wedge c \neq i\}$.
- recall(*rc*):** a measure of the ability of a prediction model to select a certain class from the data set. It is calculated by $rc_i = \frac{tp_i}{tp_i + \sum e_{ij}}$, where $i \in \{A, B, C\}$, and $j \in \{c | c \in \{A, B, C\} \wedge c \neq i\}$.
- F*-score(*F_β*):** It is the measure that combines *precision* and *recall*. It “measures the effectiveness of retrieval with respect to a user who attaches β times as much importance to recall as precision” [van Rijsbergen, 1979]. This measure is calculated by the formula $F_{\beta i} = (1 + \beta^2) \times \frac{pr_i \times rc_i}{\beta^2 \times pr_i + rc_i}$. F_1 , F_2 , and $F_{0.5}$ are the commonly used *F* measures.
- balanced error rate(*ber*):** the average wrong classification errors on each class. It is defined as $ber = \frac{\sum e_i}{3} = \frac{\sum 1 - r_i}{3}$, where $i \in \{A, B, C\}$.

Liu Yuee
(Thesis)

MATRIZ DE CONFUSIÓN – CÁLCULOS DERIVADOS

graficas.txt - WordPad

Archivo Edición Ver Insertar Formato Ayuda

#	tpi	ejk	accuracy	error_rate	pre1	pre2	pre3	pre4	pre5	pre6	pre7	pre8	rec1	rec2	rec3	rec4	rec5	rec6	rec7	rec8
1	1	90600	0.000011	6.999896	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	1	90600	0.000011	6.999896	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	37	90564	0.000408	6.996902	1.00	1.00	1.00	1.00	0.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	59	90542	0.000651	6.995154	1.00	1.00	1.00	1.00	0.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00
5	192	90409	0.002119	6.983156	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00
6	316	90285	0.003488	6.972066	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.01	1.00
7	726	89875	0.008013	6.935713	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.01	0.00	0.01	0.02	0.00	0.01	0.01	1.00
8	1098	89503	0.012119	6.903266	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.01	0.01	0.01	0.03	0.00	0.02	0.01	1.00
9	1899	88702	0.020960	6.834289	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.02	0.01	0.03	0.05	0.00	0.04	0.02	1.00
10	2723	87878	0.030055	6.763821	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.02	0.02	0.03	0.07	0.01	0.05	0.03	1.00
11	3959	86642	0.043697	6.661747	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.03	0.03	0.05	0.10	0.01	0.07	0.05	1.00
12	5207	85394	0.057472	6.560502	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.04	0.04	0.07	0.13	0.01	0.09	0.06	1.00
13	7184	83417	0.079293	6.398793	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.06	0.05	0.10	0.17	0.02	0.12	0.09	1.00
14	9100	81501	0.100440	6.245850	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.07	0.07	0.12	0.20	0.03	0.15	0.11	1.00
15	11617	78984	0.128222	6.051691	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.08	0.09	0.16	0.25	0.04	0.19	0.14	1.00
16	14035	76566	0.154910	5.867986	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.00	0.10	0.11	0.19	0.28	0.05	0.22	0.17	1.00
17	17303	73298	0.190980	5.621469	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.01	0.12	0.15	0.24	0.33	0.07	0.26	0.21	1.00
18	20237	70364	0.223364	5.402368	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.01	0.14	0.18	0.29	0.37	0.09	0.30	0.24	1.00
19	23823	66778	0.262944	5.132207	1.00	1.00	1.00	0.99	1.00	1.00	1.00	0.01	0.16	0.22	0.34	0.41	0.12	0.34	0.28	1.00
20	27191	63410	0.300118	4.882511	1.00	1.00	0.99	0.99	1.00	1.00	1.00	0.01	0.18	0.26	0.39	0.45	0.15	0.37	0.32	1.00
21	31088	59513	0.343131	4.588421	1.00	1.00	0.99	0.99	1.00	1.00	1.00	0.01	0.21	0.30	0.45	0.49	0.18	0.41	0.36	1.00
22	34566	56035	0.381519	4.321402	1.00	1.00	0.99	0.98	1.00	1.00	1.00	0.02	0.24	0.34	0.50	0.52	0.23	0.45	0.40	1.00
23	38542	52059	0.425404	4.013475	1.00	1.00	0.98	0.98	1.00	0.99	1.00	0.02	0.27	0.39	0.56	0.56	0.28	0.49	0.44	1.00
24	41980	48621	0.463350	3.742187	1.00	1.00	0.98	0.98	1.00	0.99	1.00	0.03	0.30	0.43	0.61	0.59	0.34	0.52	0.48	1.00
25	45692	44909	0.504321	3.445233	1.00	1.00	0.97	0.97	1.00	0.99	1.00	0.03	0.34	0.47	0.65	0.62	0.40	0.55	0.51	1.00
26	48772	41829	0.538316	3.197303	1.00	1.00	0.97	0.97	1.00	0.99	1.00	0.04	0.38	0.51	0.69	0.65	0.45	0.58	0.54	1.00
27	52053	38548	0.574530	2.931671	1.00	1.00	0.97	0.96	1.00	0.99	1.00	0.05	0.42	0.55	0.73	0.67	0.51	0.61	0.57	1.00
28	54910	35691	0.606064	2.698195	1.00	1.00	0.96	0.96	1.00	0.98	1.00	0.06	0.46	0.59	0.76	0.70	0.57	0.64	0.59	1.00
29	57880	32721	0.638845	2.456903	1.00	1.00	0.96	0.95	1.00	0.98	1.00	0.07	0.50	0.62	0.79	0.72	0.62	0.67	0.62	1.00
30	60538	30063	0.668182	2.240434	1.00	1.00	0.96	0.95	1.00	0.98	1.00	0.08	0.54	0.66	0.82	0.75	0.66	0.69	0.64	1.00
31	63154	27447	0.697056	2.031369	1.00	1.00	0.96	0.95	1.00	0.98	1.00	0.10	0.59	0.69	0.84	0.77	0.70	0.72	0.67	1.00
32	65420	25181	0.722067	1.850879	1.00	1.00	0.96	0.95	1.00	0.98	1.00	0.12	0.63	0.71	0.86	0.79	0.74	0.74	0.69	1.00
33	67567	23034	0.745764	1.684007	1.00	0.99	0.96	0.95	1.00	0.97	1.00	0.14	0.66	0.74	0.88	0.80	0.77	0.76	0.71	0.99
34	69431	21170	0.766338	1.543221	1.00	0.99	0.96	0.95	1.00	0.97	1.00	0.16	0.70	0.76	0.89	0.82	0.80	0.78	0.73	0.99
35	71151	19450	0.785322	1.415962	1.00	0.98	0.96	0.95	1.00	0.97	0.99	0.18	0.72	0.78	0.90	0.84	0.82	0.80	0.74	0.98
36	72657	17944	0.801945	1.306174	1.00	0.98	0.96	0.95	1.00	0.97	0.99	0.20	0.75	0.80	0.91	0.85	0.84	0.81	0.76	0.98
37	74044	16557	0.817254	1.203193	1.00	0.98	0.96	0.95	1.00	0.97	0.99	0.23	0.77	0.81	0.92	0.86	0.86	0.83	0.77	0.97
38	75225	15376	0.830289	1.117198	1.00	0.97	0.96	0.95	1.00	0.97	0.99	0.26	0.78	0.83	0.93	0.87	0.87	0.84	0.79	0.97

Para obtener Ayuda, presione F1

MAY

MATRIZ DE CONFUSIÓN – CÁLCULOS DERIVADOS

PIXELES BIEN Y MAL CLASIFICADOS

```
plot "graficas.txt" using 1:2 title 'tpi' with lines, \
"graficas.txt" using 1:3 title 'ejk' with lines
```

ACCURACY Y BALANCED ERROR RATE

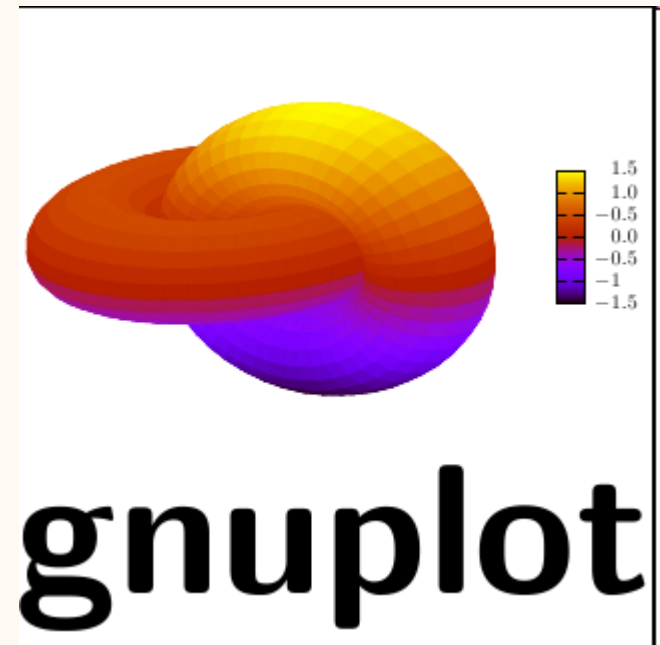
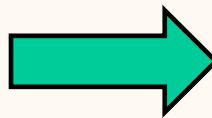
```
plot "graficas.txt" using 1:4 title 'accuracy' with lines, \
"graficas.txt" using 1:5 title 'balanced error rate' with lines
```

PRECISION para las 8 clases

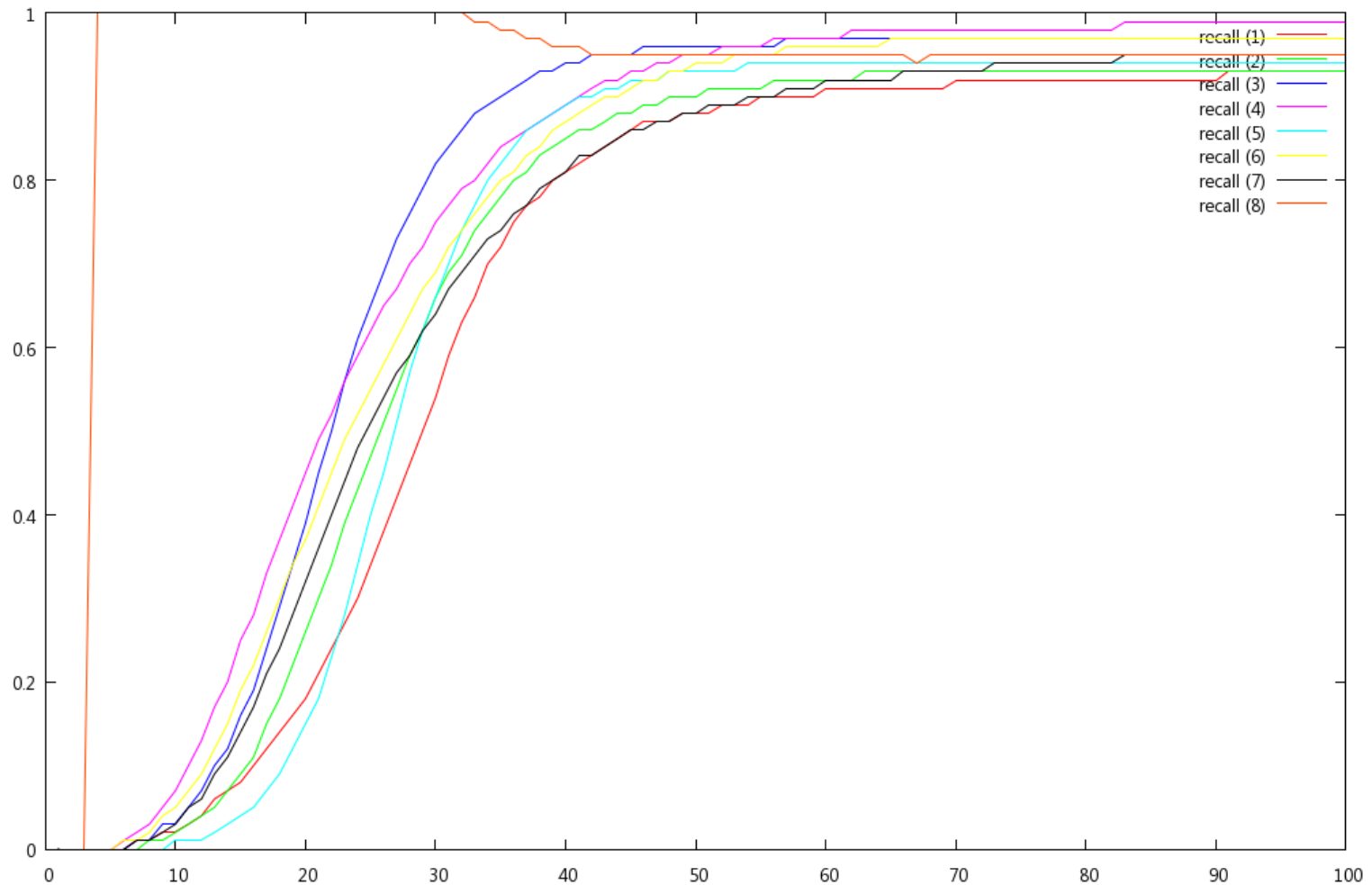
```
plot "graficas.txt" using 1:6 title 'precision (1)' with lines, \
"graficas.txt" using 1:7 title 'precision (2)' with lines, \
"graficas.txt" using 1:8 title 'precision (3)' with lines, \
"graficas.txt" using 1:9 title 'precision (4)' with lines, \
"graficas.txt" using 1:10 title 'precision (5)' with lines, \
"graficas.txt" using 1:11 title 'precision (6)' with lines, \
"graficas.txt" using 1:12 title 'precision (7)' with lines, \
"graficas.txt" using 1:13 title 'precision (8)' with lines
```

RECALL para las 8 clases

```
plot "graficas.txt" using 1:14 title 'recall (1)' with lines, \
"graficas.txt" using 1:15 title 'recall (2)' with lines, \
"graficas.txt" using 1:16 title 'recall (3)' with lines, \
"graficas.txt" using 1:17 title 'recall (4)' with lines, \
"graficas.txt" using 1:18 title 'recall (5)' with lines, \
"graficas.txt" using 1:19 title 'recall (6)' with lines, \
"graficas.txt" using 1:20 title 'recall (7)' with lines, \
"graficas.txt" using 1:21 title 'recall (8)' with lines
```



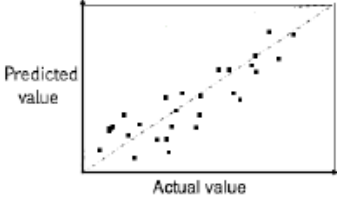
MATRIZ DE CONFUSIÓN – CÁLCULOS DERIVADOS



MATRIZ DE CONFUSIÓN – CÁLCULOS DERIVADOS

Regression performances measure cheat sheet

Damien François - v0.9 - 2009 (damien.francois@uclouvain.be)

Let $D = \{(x_i, y_i)\}$ be a set of input/output pairs and f a function such that for $i = 1..n$,	Absolute error	Robust error measures	Resampling methods
$y_i = f(x_i) + \epsilon_i$	<p>MAD Mean Absolute Deviation</p> $\frac{1}{n} \sum \epsilon_i $ <p>MAPE Mean Absolute Percentage Error</p> $\frac{1}{n} \sum \frac{ \epsilon_i }{y_i}$	<p>Median Squared error</p> $\text{median}(\epsilon_i^2)$ <p>α-trimmed MSE</p> $\frac{1}{\#I} \sum_{i \in I} \epsilon_i^2$ <p>where I is the set of residuals ϵ_i where α percents of the largest values are discarded.</p> <p>M-estimators</p> $\frac{1}{n} \sum \rho(\epsilon_i)$ <p>where ρ is a non-negative function with a minimum in 0, like the parabola, the Hubber function, or the bisquare function.</p>	<p>LOO - Leave-one-out: build the model on $n - 1$ data elements and test on the remaining one. Iterate n times to collect all ϵ_i and compute mean error.</p> <p>X-Val - Cross validation. Randomly split the data in two parts, use the first one to build the model and the second one to test it. Iterate to get a distribution of the test error of the model.</p> <p>K-Fold - Cut the data into K parts. Build the model on the K-1 first parts and test on the Kth one. Iterate from 1 to K to get a distribution of the test error of the model.</p> <p>Bootstrap - Draw a random subsample of the data with replacement. Compute the error on the whole dataset minus the training error of the model and iterate to get a distribution of such values. The mean of the distribution is the optimism. The bootstrap error estimate is the training error on the whole dataset plus the optimism.</p>
<p>Squared error</p> <p>SSE Sum of Squared Errors, or RSS Residual Sum of Squares</p> $\sum \epsilon_i^2$ <p>MSE Mean Squared Error</p> $\frac{1}{n} \sum \epsilon_i^2$ <p>RMSE Root Mean Squared Error</p> $\sqrt{\frac{1}{n} \sum \epsilon_i^2}$ <p>NMSE Normalised Mean Squared Error</p> $\frac{SSE}{\text{var}(\{y_i\})}$ <p>where var is the empirical variance in the sample.</p> <p>R-squared</p> $1 - \frac{SSE}{\text{var}(y_i)}$ <p>where var is the empirical variance in the sample</p>	<p>Predicted error</p> <p>PRESS Predicted Residual Sums of Squares</p> $\frac{1}{n} \ \text{diag}(XX^T)(XX^T - I)Y\ _2^2$ <p>where X is a matrix built by stacking the x_i in rows. Y is the vector of y_i</p> <p>GCV Generalised Cross Validation</p> $\frac{\frac{1}{n} \ (I - X(X^T X + nI)^{-1} X^T)Y\ _2^2}{(\frac{1}{n} \text{Trace}(I - X(X^T X + nI)^{-1} X^T))^2}$ <p>where X is a matrix built by stacking the x_i in rows. Y is the vector of y_i</p>	<p>Graphical tool</p> <p>Plot of predicted value against actual value. A perfect model places all dots on the diagonal.</p> 	
	<p>Information criteria</p> <p>AIC Akaike Information Criterion</p> $n \log MSE + 2k$ <p>where k is the number of parameters in the model</p> <p>BIC Bayesian Information Criterion</p> $n \log MSE + k \cdot \log n$ <p>where k is the number of parameters in the model</p>		

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

DISTANCIA DE MAHALANOBIS

```
proceso - Microsoft Visual Studio
File Edit View Project Build Debug Tools Window Community Help
Debug Win32
Proceso.c*
(Global Scope)
main(int argc, char ** argv)
imagen_campo_archivo = fopen (imagen_campo_archivo_nombre, "r");
for (i=0; i<imagen_original[0]->height; i++)
  for (j=0; j<imagen_original[0]->width; j++)
    fscanf (imagen_campo_archivo, "%d", &imagen_campo[i][j]);
fclose (imagen_campo_archivo);

// Lee las muestras de las clases del archivo .SIG
archivoSIG=ehfa_FileOpen(muestras_sig,"r",erdinit,&error);
lecturSIG=esig_SignatureArrayReadImagine (archivoSIG, EHFA_ROOT_NODE, &error);
aCalidad = fopen (acsig_dat , "w");
fprintf (aCalidad, "CONJUNTO DE ENTRENAMIENTO \n");
for (i=0; i<NUM_CLASES; i++)
{
  signatureSIG=ESIG_SIGNATURE (lecturSIG,i,&error);
  for (j=0; j<NUM_CAPAS; j++)
  {
    cto_muestras[i][j]=signatureSIG->statstack->allStats[j]->statistics->mean;
    ESIG_BAND_COV(signatureSIG,i,j);
    fprintf (aCalidad, "Media SIG: %g \n", cto_muestras[i][j]);
  }
}
fclose (aCalidad);
```

signatureSIG = muestras de una clase concreta

vector de medias de la clase (capas)

matriz de covarianza de la clase (i, j son la posición en la matriz)

Properties: main VCCodeFunction

(Name)	main
File	d:\archivos de pr...
FullName	main
IsInjected	False
IsInline	False
IsOverloaded	False
IsSealed	False
IsTemplate	False
TypeString	int

(Name)
Sets/returns the name of the object.

Ln 289 Col 1 Ch 1 INS

$$\text{Dist} = \sqrt{(x-u)^T * S^{-1} * (x-u)}$$

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

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 (lógica difusa).**
- **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.**

CLASIFICACIÓN DIFUSA

TRABAJOS FUTUROS

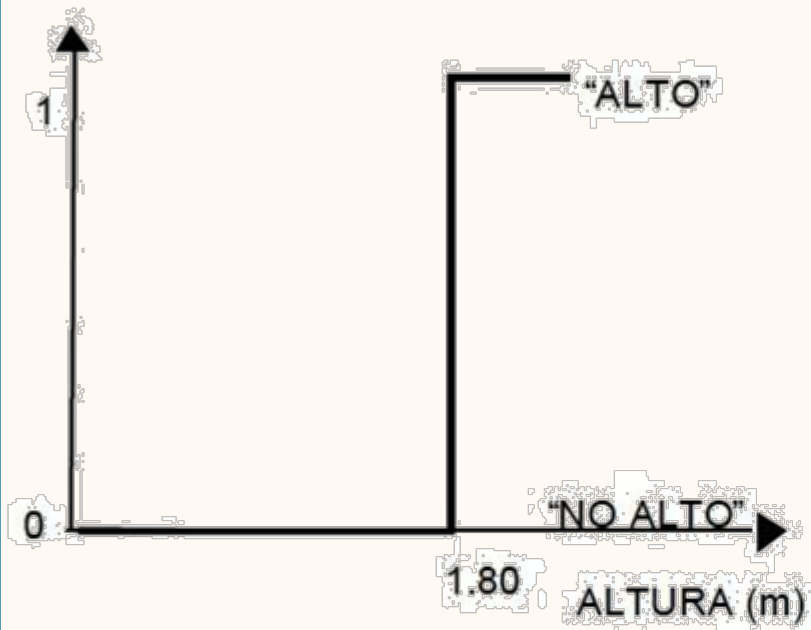
-Desarrollo de nuevas versiones del algoritmo de ACA con una configuración de estados y reglas del autómata celular distintas para personalizar aun mas el proceso de clasificación (lógica difusa).



CLASIFICACIÓN DIFUSA

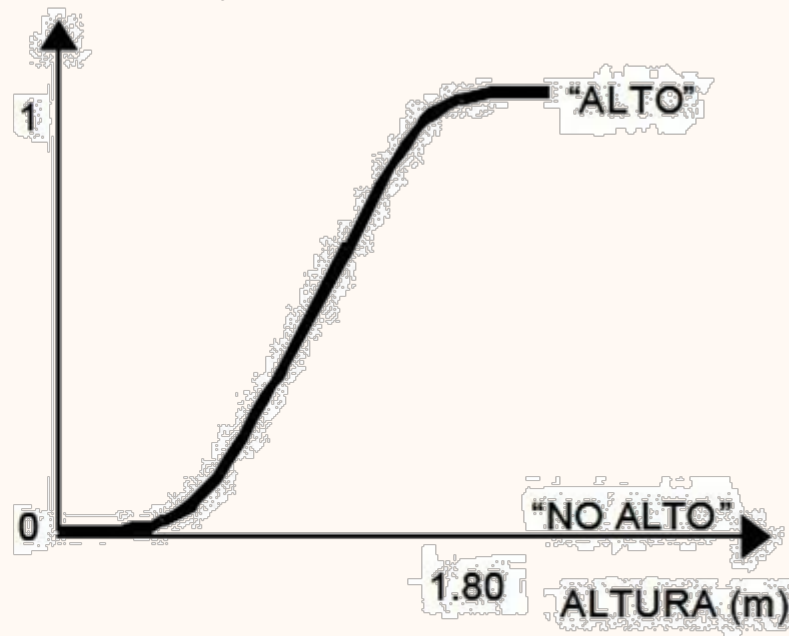
FUNDAMENTOS DE LÓGICA DIFUSA

VISIÓN DE LA LÓGICA CLÁSICA



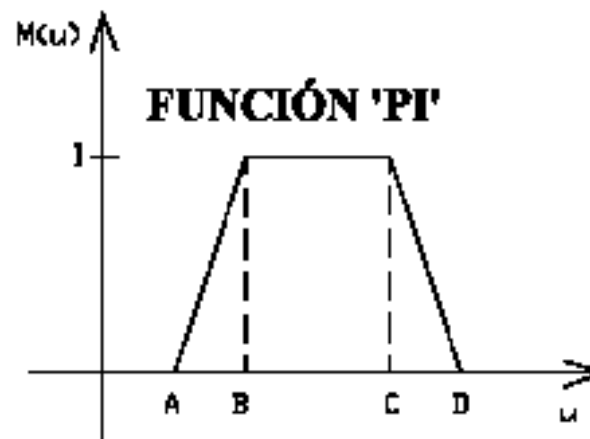
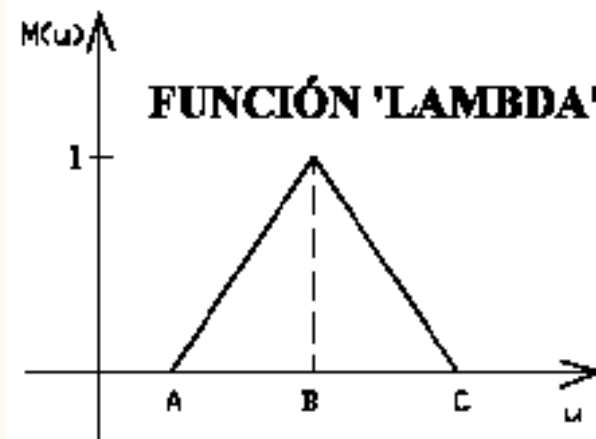
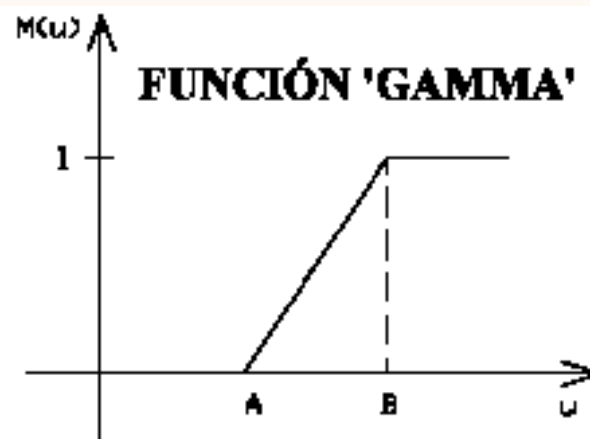
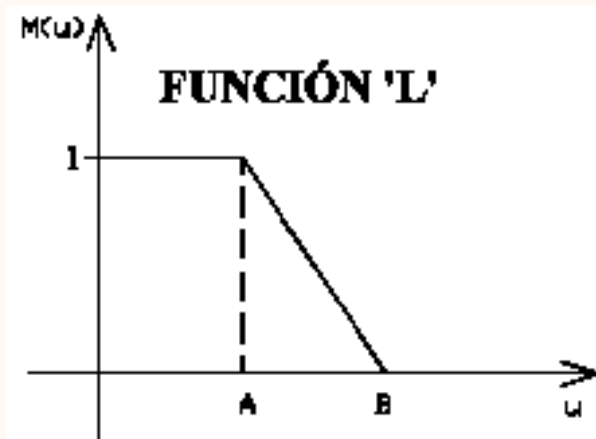
$$\mu_A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases}$$

VISIÓN DE LA LÓGICA DIFUSA



$$\text{sop}(x) = \{x \in U \mid \mu_A(x) > 0\}$$

FUNCIÓN DE PERTENENCIA EN LÓGICA DIFUSA



MATRIZ DE CONFUSIÓN DIFUSA

		Referencia						Precisión del usuario			
		Arboles	Cuerpos de agua	Pasto de altura media	Pasto natural	Pastos cortos	Zonas sin cobertura vegetal	Determinista		Fuzzy	
Clasificación	Arboles	126	0.3	1.1	0.0	2.0	0.0	126/133	94 %	129/133	97 %
	Cuerpos de agua	0.0	8	0.0	0.0	0.0	0.0	8/8	100 %	8/8	100 %
	Pasto de altura media	0.0	3.3	61	1.1	15.14	0.0	61/98	62 %	80/98	82 %
	Pastos corto poco denso	0.0	0.0	0.0	4		0.0	4/4	100 %	4/4	100 %
	Pastos cortos	0.0	0.0	13.2	0.0	18	1.1	18/34	53 %	32/34	94 %
	Zonas sin cobertura vegetal	0.0	0.0	0.0	0.0	0.0	77	77/77	100 %	77/77	100 %

Precisión del productor							
Total determinista		126/126	8/17	61/78	4/6	18/49	77/78
Porcentual determinista		99 %	44 %	77 %	67 %	36 %	97 %
Total Fuzzy		126/126	11/17	75/78	5/6	35/49	78/78
Porcentual Fuzzy		99 %	65 %	96 %	83 %	71 %	100 %

Precisión general			
Determinista		Fuzzy	
294/354	83 %	330/354	93 %

[[Ver artículo de clasificación difusa](#)]

ALGORITMO ACA DIFUSO

- Para obtener la clasificación difusa de un pixel se tienen en cuenta 2 niveles:**
- La/s clase/s del pixel (obtenida/s en SpectralACA) → **ESPECTRAL** (pixel).
 - Las clases de los vecinos del pixel (4, 8 ó 24 vecinos) → **CONTEXTUAL** (espacial).

Si el número de clases obtenido en SpectralACA es 1, el grado de pertenencia a dicha clase es muy alto.

Si el número de clases obtenido en SpectralACA es mayor que 1, nos fijamos en las clases vecinas. Por ejemplo, con 8 vecinos, si tenemos de 0 a 2 píxeles vecinos de la misma clase y el resto son distintos, el pixel actual tiene un grado de pertenencia a dicha clase bajo. Si tenemos de 2 a 6 vecinos de la misma clase y el resto son distintos, el pixel actual tiene un grado de pertenencia a dicha clase medio. Por último, si tenemos de 6 a 8 vecinos de la misma clase y el resto son distintos, el pixel actual tiene un grado de pertenencia a dicha clase alto.

ALGORITMO ACA DIFUSO

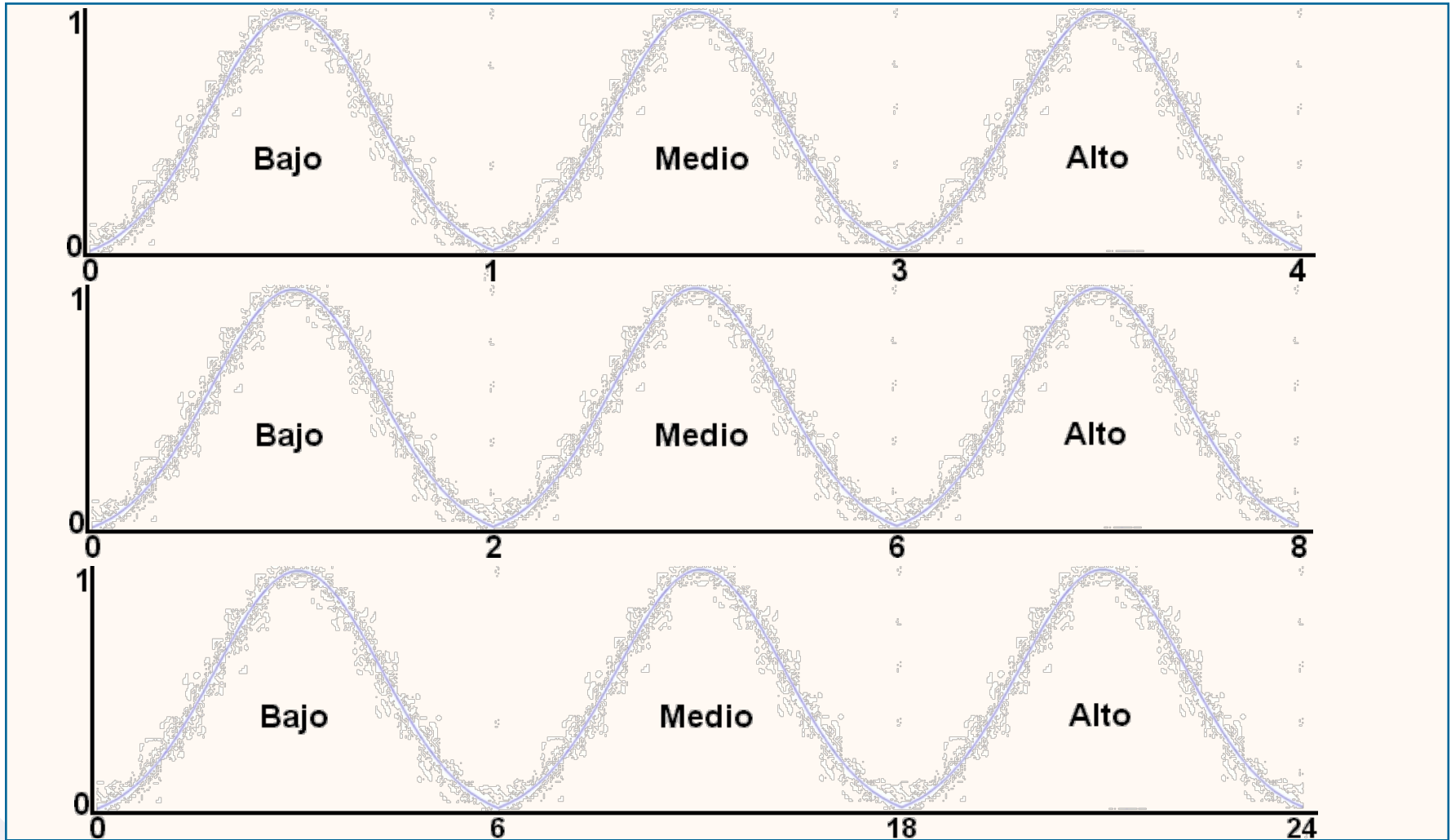
CONDICIÓN	GRADO DE PERTENENCIA
Si $SpectralClasses=1$ →	Muy alto
Si $SpectralClasses>1 \ \&\& \ x=0$ →	Bajo
Si $SpectralClasses>1 \ \&\& \ 1 \leq x \leq 3$ →	Medio
Si $SpectralClasses>1 \ \&\& \ x=4$ →	Alto
	} 4 vecinos
Si $SpectralClasses>1 \ \&\& \ 0 \leq x < 2$ →	Bajo
Si $SpectralClasses>1 \ \&\& \ 2 \leq x \leq 6$ →	Medio
Si $SpectralClasses>1 \ \&\& \ 6 < x \leq 8$ →	Alto
	} 8 vecinos
Si $SpectralClasses>1 \ \&\& \ 0 \leq x < 6$ →	Bajo
Si $SpectralClasses>1 \ \&\& \ 6 \leq x \leq 18$ →	Medio
Si $SpectralClasses>1 \ \&\& \ 18 < x \leq 24$ →	Alto
	} 24 vecinos

Donde:

$SpectralClasses$ → número de clases espectrales devueltos por SpectralACA.

x → número de vecinos con la misma clase que el pixel actual.

ALGORITMO ACA DIFUSO



Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

PROYECTO FIN DE MASTER

“Autómatas celulares aplicados a un DEM para la simulación del caudal de pantanos y ríos en función de condiciones climatológicas variables”



Proyecto Fin de Master

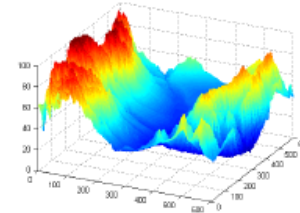
Directores:
José Antonio Piedra
Moisés Espínola Pérez

Grupo de Investigación de Informática Aplicada
Dpto. Lenguajes y Computación
Universidad de Almería

1

1. Descripción del proyecto

En este proyecto se pretende realizar una simulación, basada en autómatas celulares, del caudal de pantanos y ríos en base al volumen de precipitaciones recibidas en un intervalo de tiempo determinado. Dicha simulación se realizará sobre una imagen DEM (Digital Elevation Model, o modelo de elevación digital del terreno), es decir, una imagen de satélite donde se almacena en cada píxel su altura correspondiente. A continuación se muestra un ejemplo de imagen DEM:



Las características principales del algoritmo serán las siguientes:

Condiciones iniciales:

- Partimos de unas condiciones iniciales (volumen de la precipitación en litros por metro cuadrado, cantidad de agua ya existente en el pantano/rio, etc).

Configuración del autómata celular:

- Cada píxel de la imagen DEM corresponderá con una celda del autómata celular.
- Cada celda del autómata celular tendrá dos estados. El primero será la altura a la que se encuentra el píxel de la imagen DEM, y el segundo la el volumen de agua que existe por encima.

Simulación de fluidos:

- En cada iteración del autómata celular, caerá sobre cada píxel de la imagen DEM el volumen de precipitación por segundo establecido en las condiciones iniciales.
- La función de transición, a través de las reglas del autómata celular, deberá ir pasando el volumen correspondiente de agua de cada celda a su vecino con menor elevación (ya que el agua tiende a bajar).
- Cuando el agua vaya llegando a píxeles menos elevados (con vecinos con elevación más alta), se acumula la altura (altura del píxel más volumen de agua acumulada).
- Así sucesivamente, hasta que deje de llover.

Resultados obtenidos:

- Para finalizar habrá que calcular el caudal acumulado del pantano o río (altura actual menos altura real de los píxeles), y marcarlo en azul en la imagen de satélite.

2

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

CLASIFICACIÓN DIFUSA

TRABAJOS FUTUROS

-Utilizar mejora no sólo en clasificación contextual sino también en clasificación textural.



CLASIFICACIÓN TEXTURAL

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

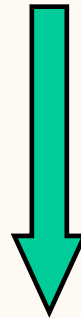
Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

CLASIFICACIÓN DIFUSA

TRABAJOS FUTUROS

- **Utilización de agentes de software para reducir el coste computacional, recorriendo varias regiones de la imagen en paralelo.**



AGENTES SOFTWARE

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

Imágenes de satélite y autómatas celulares: nuevas técnicas de clasificación

- **TELEDETECCIÓN**
- **CLASIFICACIÓN CON AUTÓMATAS CELULARES**
- **RESULTADOS OBTENIDOS**
- **TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN**
- **TRABAJOS FUTUROS – NUEVOS CLASIFICADORES**
- **TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA**
- **TRABAJOS FUTUROS – PROYECTO FIN DE MASTER**
- **TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL**
- **TRABAJOS FUTUROS – PARALELIZACIÓN**
- **CONCLUSIONES**

SATELLITE IMAGE CLASSIFICATION BASED ON A CELLULAR AUTOMATA APPROACH A Hierarchical and Contextual Classification Algorithm

M. Espínola¹, J.A. Piedra¹, R. Ayala¹, I. Irberna¹, S. Leguizamón² and M. Menenti³

¹Applied Computing Group, University of Almería, Almería, Spain

²Regional Faculty, National Technological University of Mendoza, Mendoza, Argentina

³Aerospace Engineering Optical and Laser Remote Sensing

{moez.espinola, jpiedra, rayala, iis.tribarne}@ual.es, satarmino.leguizamoni@pymedi.com.ar, mmomenenti@fmad.fcit

Keywords: image classification, pattern recognition, remote sensing, cellular automata.

Abstract: The satellite image classification is one of the most important techniques used in remote sensing for the computerized analysis and pattern recognition of satellite data that facilitates the researchers the automated interpretation of a great deal of information. There are currently many types of classification algorithms, such as minimum distance, parallelepiped and maximum likelihood classifiers, but none is fault-tolerant in terms of quality. This paper studies the development of a new classification algorithm based on cellular automata called ACA (Algorithm based on Cellular Automata), which offers a hierarchical classification divided into levels of reliability with a final quality optimized through contextual techniques. There few previous works related to satellite image classification with cellular automata.

1 INTRODUCTION

Remote sensing is the most relevant science that allows us the acquisition of information about the surface of the land and environmental information values without having actual contact with the area being observed (Carrasco and Hasto, 2010). This science can be used in many environmental applications, helping to solve and improve the problems derived from them. Examples of remotely sensed applications include soil quality studies, water resources searching, environmental protection or meteorology simulations, among others.

The satellite image classification is one of the most important techniques used in remote sensing that helps on interpreting a great deal of information contained in the spectral bands, grouping together the image pixels in a finite number of classes, basing on the spectral values of the bands (Rao, 2001). The information obtained by the satellite sensors as digital levels is changed into a categorical scale way to interpret by the analyst experts. The resulting classified satellite image is essentially a thematic map of the original image and pixels belonging to the same class will have similar spectral characteristics.

The classification algorithms work really well in non-noisy satellite images and if the spectral properties of the pixels determine the classes sufficiently well. However, if the images are blurred with a gaussian impulse-type noise or there are essential changes in the pixels properties, the resulting image may have lots of tiny areas (often a pixel) which are misclassified. To sort out this classification error, we can use contextual information taking into account not only its spectral values but also its surrounding pixels. There are several contextual classification algorithms which use average values or texture description to improve the spectral classification, but but none is full-reliable in terms of quality. This paper studies the development of a new classification algorithm based on cellular automata that offers a hierarchical classification divided into levels of reliability and improves the final quality. The rest of the paper is structured as follows. Section 2 describes the classical classification algorithms problems. Section 3 describes the basic aspects of cellular automata. In section 4 we focus in the use of cellular automata to classify satellite images (ACA algorithm). Section 5 shows the results and conclusions. Finally, in section 6, we finish the paper exposing future work.

2 SATELLITE IMAGE CLASSIFICATION PROBLEMS

In the literature, there are different procedures to classify satellite images, and the image classification techniques can be divided into two main categories: supervised and unsupervised classification algorithms. The use of supervised or unsupervised methods in the classification process depends on the analyst expert knowledge of the satellite image study area (Ayala et al, 2002).

In an unsupervised classification algorithm, the analyst only specifies the number of classes, and the algorithm groups the satellite image pixels based solely on the numerical information in the data. In these algorithms, the analyst has not to know the zone to study. There are many unsupervised classification algorithms, like the Iterated algorithm, K-means, Leader, MaxMin or Neural Model unsupervised. The classification accuracy obtained by these methods are less reliable than the results offered by the supervised classification algorithms, because the computer groups the image pixels without taking into account the expert knowledge of the study area.

In a supervised classification, the analyst selects samples of the different classes to identify the pixels in the image (training group), so the algorithm can make a more precise classification based on the human expert knowledge. Therefore, in this method the analyst knowledge of the study area determines the quality of the training set. The computer uses an algorithm to compare each pixel in the image to these signatures chosen by the expert. The pixels are labelled as the class most closely resembles digitally. There are several types of statistics based supervised classification algorithms. Some of the more popular ones are parallelepiped, minimum distance, maximum likelihood, fuzzy supervised, neural model and Mahalanobis distance, among others. Supervised classification algorithms offer a better classification results than unsupervised classification.

In spite of the great number of image classification algorithms that exist, none is full-reliable in terms of quality. In this paper we focus in the classification process with these three classical algorithms: minimum distance, parallelepiped and maximum likelihood classifiers. Figure 1 shows the classification process with these three classical algorithms from the viewpoint of feature space and taking into account three different classes in the satellite image.

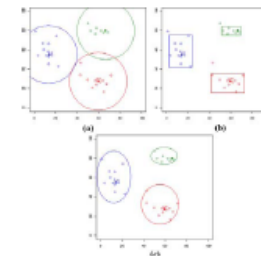


Figure 1: (a) Minimum distance, (b) parallelepiped and (c) maximum likelihood

The minimum distance algorithm assigns each image pixel to the class with which there is less spectral euclidean distance with respect its centroid, for all bands involved in the classification process. This algorithm has the disadvantage of being prone to commission errors (assign a pixel to a wrong class) and not considers the variance of each one of the classes.

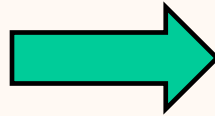
The parallelepiped algorithm assigns each image pixel to a class if its spectral levels are within the spectral domain area of that class, for all bands involved in the classification process. This algorithm performs better than minimum distance algorithm, but it has the disadvantage that some pixels may be unclassified after the process or misclassified in several classes.

The maximum likelihood algorithm considers the mean and standard deviation of each class to calculate the probability that a given image pixel belongs to each one of the classes. This algorithm is more reliable than previous algorithms, but requires a large computational cost.

If we add to the above disadvantages the presence of noise in the satellite image and heterogeneity in the geographical distribution of the classes, the results obtained by these classical supervised classification algorithms decrease considerably. Most of these disadvantages can be overcome with the use of cellular automata that use contextual information taking into account not only the pixel spectral values but also its surrounding pixels.

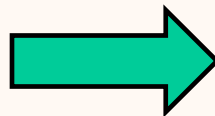
RESUMEN

**ARTÍCULO
DE REVISTA**



- TRABAJOS FUTUROS – MATRIZ DE CONFUSIÓN
- TRABAJOS FUTUROS – NUEVOS CLASIFICADORES
- TRABAJOS FUTUROS – CLASIFICACIÓN DIFUSA

**ARTÍCULOS
CONGRESOS**



- TRABAJOS FUTUROS – PROYECTO FIN DE MASTER
- TRABAJOS FUTUROS – CLASIFICACIÓN TEXTURAL
- TRABAJOS FUTUROS – PARALELIZACIÓN

RUEGOS Y PREGUNTAS

AGRADECIMIENTOS

**¡MUCHAS GRACIAS
POR LA ATENCION!**

