



UNIVERSITAT POLITÈCNICA  
DE CATALUNYA  
BARCELONATECH



Universitat Politècnica de Catalunya

Escola Tècnica Superior d'Enginyeria de Telecomunicacions de Barcelona Signal Theory  
and Communications Department

# Trabajo Final de Grado

Por Cristian Cuadrado Conde

## PolSAR image classification with Binary Partition Tree

Supervisores: Prof. Philippe Salembier Clairon  
Prof. Carlos López Martínez

# Resumen:

*Binary Partition Tree* (BPT) es una herramienta que permite una representación multi-escala de imágenes basada en la fusión de regiones de la imagen, en la que cada nodo del BPT, representa una región. Hoy en día, se está aplicando esta herramienta en los sistemas de teledetección conocidos como PolSAR, con el objetivo representar elementos de la imagen PolSAR con diferentes niveles de detalle.

El objetivo de este proyecto es estudiar nuevas técnicas en el uso de *Binary Partition Tree's* para mejorar la clasificación de objetos en imágenes PolSAR. Se desarrollaran distintas técnicas y se comparan los resultados con el estado del arte. Por último, se exponen algunas conclusiones y se presentan y comentan algunas posibles líneas de trabajo futuras.

Palabras clave: Visión por computador, Aprendizaje automático, PolSAR, Binary partition Tree.

# Resum:

*Binary Partition Tree*(BPT), és una eina que permet una representació multiescala d'imatges basada en la fusió de regions de la imatge, en la que cada node del BPT, representa una regió. Avui dia, s'està aplica'n aquesta eina en els sistemes de teledetecció coneguts com a PolSAR, amb l'objectiu de representar elements de l'imatge PolSAR amb diferents nivells de detall.

L'objectiu d'aquest projecte és estudiar noves tècniques en l'ús de *Binary Partition Tree's* per millorar la classificació d'objectes en imatges PolSAR. Es desenvolupen diferents tècniques i es comparen els resultats obtinguts amb l'estudi de l'art. Per acabar, s'exposen algunes conclusions i es presenten i comenten algunes possibles línies de treball futures.

Paraules clau: Visió per computador, Aprenentatge automàtic, PolSAR, Binary partition Tree.

# Summary:

*Binary Partition Tree* (BPT) is a tool that allows a multiscale image representation based on the merging of regions of the image, in which each BPT node represents a region. Today, this tool is being applied to remote sensing systems known as PolSAR, with the aim of representing elements of the PolSAR image with different levels of detail.

The objective of this project is to study new techniques in the use of *Binary Partition Tree* to improve the classification of objects in PolSAR images. Different techniques will be developed and the results will be compared with the state of the art. Finally, some conclusions are presented and some possible lines of future work are discussed.

Key words: Computer vision, Machine learning, PolSAR, Binary partition Tree.

# Agradecimientos

En primer lugar a mis profesores Philippe Salembier Clairon y Carlos Lopez Martinez por toda su ayuda para tirar adelante el proyecto. A Ferran Pérez e Ismael Vivó, por el compañerismo durante la tesis. A la gente del edificio Omega por convertir Campus Nord en una segunda casa. A los profesores que consiguieron meterme algo útil en la cabeza. A toda la gente que he ido conociendo estos años porque si no la cosa pierde la gracia. Y para terminar, a mi familia, a mis padres por preocuparse que tuviera estudios, a mi hermano pequeño por demasiadas cosas para mencionarlas y a mi abuela con la que viví todos estos años.

## Creación y revisión del documento

<b>Revisión</b>	<b>Fecha</b>	<b>Objetivo</b>
1	13/04/2017	Creación documento
2	25/04/2017	Revisión del documento
3	05/05/2017	Revisión del documento
4	10/05/2017	Revisión del documento
5	15/05/2017	Revisión del documento

Implicados en el proyecto:

<b>Nombre</b>	<b>Correo</b>
Cristian Cuadrado Conde	c.cuadrado91@gmail.com
Philippe Salembier Clairon	philippe.salembier@upc.edu
Carlos López-Martínez	carlos.lopez@tsc.upc.edu

**Escrito por**

**Revisado y  
aprobado por**

<b>Fecha:</b>	10/05/2017	<b>Fecha:</b>	12/05/2017
<b>Nombre:</b>	Cristian Cuadrado	<b>Nombre</b>	Philippe Salembier Carlos López
<b>Cargo:</b>	Autor del proyecto	<b>Cargo</b>	Supervisores

# Contents

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Visión general . . . . .	1
1.2	Objetivos del proyecto . . . . .	1
1.3	Plan de trabajo . . . . .	1
<b>2</b>	<b>Introducción a PolSAR, BPT y clasificación empleando algoritmos de clasificación supervisada</b>	<b>3</b>
2.1	Definición de los sistemas PolSAR . . . . .	3
2.2	Polarimetría SAR . . . . .	3
2.2.1	Dispersión Polarimétrica . . . . .	3
2.2.2	Representación de imágenes PolSAR . . . . .	4
2.3	Distribución de dispersión . . . . .	5
2.4	Matrices de covarianza y coherencia de muestras . . . . .	6
2.5	Descriptores PolSAR . . . . .	6
2.5.1	Matriz de coherencia $\underline{T}$ . . . . .	6
2.5.2	Descomposición de Hoekman . . . . .	7
2.5.3	Descomposición en autovalores, H, A y alpha . . . . .	8
2.6	Binary Partition Tree . . . . .	9
2.6.1	Representación BPT . . . . .	9
2.6.2	Conectividad de la imagen . . . . .	10
2.6.3	Medidas de similitud . . . . .	10
2.7	Algoritmos de clasificación supervisados . . . . .	10
2.7.1	Support Vector Machine (SVM) . . . . .	11
2.7.2	Árbol de decisión . . . . .	12
2.8	Evaluación de los resultados, Fscore . . . . .	12
2.9	Datos de entrenamiento y test . . . . .	13
2.9.1	Datos de entrenamiento . . . . .	13
2.9.2	Datos de test . . . . .	13
2.10	Normalización . . . . .	14

2.11	Estado del arte clasificación PolSAR . . . . .	14
<b>3</b>	<b>Algoritmo de clasificación con BPT</b>	<b>16</b>
3.1	Idea principal . . . . .	16
3.2	BPT para mejorar la clasificación . . . . .	16
3.2.1	Clasificación de los nodos . . . . .	16
3.2.2	Descriptores . . . . .	17
3.3	Ramas del BPT . . . . .	17
3.3.1	Representación de las ramas . . . . .	17
3.3.2	Mezcla de clases . . . . .	17
3.4	Análisis de los nodos . . . . .	18
3.4.1	Identificación de la clase del nodo . . . . .	18
3.4.2	Representación de la clasificación de los nodos . . . . .	19
3.5	Uso de la probabilidad a posteriori para la clasificación . . . . .	20
3.5.1	Uso de los N primeros componentes . . . . .	20
3.5.2	Probabilidad condicionada en función del área . . . . .	21
<b>4</b>	<b>Resultados</b>	<b>26</b>
4.1	Clasificación a nivel de píxel . . . . .	26
4.1.1	<b>Selección de descriptores</b> . . . . .	26
4.1.2	<b>Promediado de la imagen</b> . . . . .	27
4.1.3	Resultados de la clasificación de píxeles . . . . .	28
4.2	Clasificación mediante BPT . . . . .	29
4.2.1	<b>N primeros componentes</b> . . . . .	29
4.2.2	<b>Probabilidad condicionada en función del área</b> . . . . .	31
4.2.3	<b>Análisis de los resultados</b> . . . . .	32
<b>5</b>	<b>Conclusiones y posible líneas de trabajo futuro</b>	<b>37</b>
5.1	Conclusiones . . . . .	37
5.2	Posibles líneas de trabajo futuro . . . . .	37

## List of Figures

1	Eje de coordenadas de la onda incidente y dispersa. Imagen obtenida de la referencia[1] . . . . .	4
2	Representación RGB. . . . .	5
3	Representación de una celda de resolución con múltiples objetivos[1] . . . . .	5
4	Representación del BPT . . . . .	9
5	Cada una de las hojas representa un píxel . . . . .	9
6	Los nodos representan la información de las hojas que los forman. . . . .	10
7	A la izquierda ejemplo de conectividad cuatro. A la derecha ejemplo de conectividad ocho . . . . .	10
8	Separación entre clases. Las fronteras representadas en amarillo y negro son ejemplos de propuestas del sistema mientras busca la separación óptima, se puede ver que la frontera entre puntos de la dos clases se va modificando. . . . .	11
9	Representación de la estructura de un árbol. La base de datos utilizada para representar este árbol es <i>fisheriris</i> , extraída del repositorio de Matlab . . . . .	12
10	Imágenes de test. . . . .	14
11	Representación de la rama de la hoja A, de la cual forman parte los nodos A, F, H y I. El nodo F es padre del nodo A, así mismo los nodos F y G son hijos de H. . . . .	16
12	Representación del porcentaje de patata en cada nodo. El color amarillo representa 100% de patata. . . . .	17
13	Clasificación comparada con el porcentaje de patata del nodo. . . . .	19
14	Representación de la probabilidad a posteriori. . . . .	19
15	A la izquierda la probabilidad a posteriori de la rama. A la derecha la representación de los N primeros valores con N=15. . . . .	20
16	Representación de la importancia de los descriptores. . . . .	21
17	Representación del incremento del área en logaritmo base 10. . . . .	21
18	Error por cantidad de intervalos. . . . .	22
19	Representación de los nodos seleccionados. . . . .	23
20	Resultado final. . . . .	24
21	Importancia de los intervalos. . . . .	24
22	Clasificación de píxel. . . . .	29
23	Comparación de la clasificación con N primeros componentes. . . . .	30
24	Comparación de las clasificaciones sobre las imágenes de test. . . . .	32

25	Clasificación de la imagen 3. . . . .	32
26	Píxel seleccionado . . . . .	33
27	Representación de la rama . . . . .	33
28	Píxel seleccionado . . . . .	33
29	Representación de la rama . . . . .	34
30	Píxel seleccionado . . . . .	34
31	Representación de la rama . . . . .	34
32	Píxel seleccionado . . . . .	35
33	Representación de la rama . . . . .	35
34	Píxel seleccionado . . . . .	35
35	Representación de la rama . . . . .	36

## List of Tables

1	Fscore por porcentaje de elementos de la misma clase. . . . .	18
2	Fscore considerando como límite superior 90%. . . . .	18
3	Fscore considerando como límite superior 80%. . . . .	19
4	Probabilidad condicionada en función del área, tabla 1 . . . . .	24
5	Probabilidad condicionada en función del área, tabla 2 . . . . .	24
6	Fscore del descriptor. . . . .	26
7	Fscore al excluir el descriptor. . . . .	26
8	Fscore del modelo con matriz de coherencia T y Entropía, Anisotropía y ángulo alpha. . . . .	27
9	Comparación de los modelos. . . . .	27
10	Fscore de la clasificación de los datos del modelo y de los datos de test. . . . .	27
11	Fscore por cada filtro. . . . .	28
12	Fscore de las imágenes de test. . . . .	28
13	Fscore de la clasificación con N primeros componentes comparado con el resultado de la clasificación de píxel. . . . .	30
14	Clasificación utilizando probabilidad condicionada en función del área comparado con el resultado de la clasificación de píxel. . . . .	31

# Capítulo 1

## 1 Introducción

Este capítulo da una visión general del proyecto y su contexto, los objetivos principales y el plan de trabajo.

### 1.1 Visión general

*Binary Partition Tree* (BPT) es una herramienta que permite una representación multiescala de imágenes basada en la fusión de regiones de la imagen, en la que cada nodo del BPT, representa una región. Hoy en día, en los sistemas de teledetección conocidos como PolSAR (*Polarimetric Synthetic Aperture Radar*) se está empezando a aplicar esta técnica [1] [2], con el objetivo representar elementos de la imagen PolSAR con diferentes niveles de detalle.

El objetivo de este proyecto es estudiar nuevas técnicas en el uso de *Binary Partition Tree's* para mejorar la clasificación de objetos en imágenes PolSAR. Se estudiara el actual estado del arte para realizar una clasificación de los píxeles de la imagen empleando descriptores PolSAR y técnicas de aprendizaje automático para clasificación. Una vez clasificados los píxeles, se emplearan diferentes métodos, utilizando el BPT, para mejorar los resultados obtenidos.

### 1.2 Objetivos del proyecto

Los objetivos principales del proyecto son:

- 1.- Estudiar descriptores para la clasificación de imágenes PolSAR y algoritmos de clasificación. Utilizar el BPT para mejorar el resultado de dicha clasificación.
- 2.- Analizar, evaluar y comparar las técnicas desarrolladas con el actual estado del arte.
- 3.- Conclusiones y posibles líneas de trabajo futuro.

### 1.3 Plan de trabajo

El trabajo realizado se puede dividir en diferentes etapas:

La primera parte consistió en documentar el actual estado del arte en descriptores de imágenes PolSAR, algoritmos de clasificación y estudiar el funcionamiento del BPT. Esta tarea puede verse en el capítulo 2 de este documento.

La segunda parte consistió en realizar la clasificación a nivel de píxel y escoger los parámetros y descriptores adecuados para realizar la clasificación. Esta parte puede verse en el capítulo 4.

La tercera parte consistió en probar diferentes técnicas para utilizar el BPT para tratar de mejorar de resultar el resultado de la clasificación a nivel de píxel, la discusión de las técnicas propuestas se puede ver en el capítulo 3.

La cuarta parte del proyecto consistió en una recopilación de los resultados y discusión de los resultados. Puede verse en el capítulo 4.

## Capítulo 2

### 2 Introducción a PolSAR, BPT y clasificación empleando algoritmos de clasificación supervisada

#### 2.1 Definición de los sistemas PolSAR

Los sistemas *Synthetic Aperture Radar* SAR [5] son radares de apertura sintética capaces de obtener información de alta resolución espacial. Uno de los mayores atractivos de un sistema de teledetección activo de microondas, como es el caso de los sistemas SAR, respecto otros sistemas de teledetección de imagen, es que producen su propia fuente de iluminación. Eso implica que su funcionamiento es independiente de la mayoría de condiciones atmosféricas, tales como el ciclo del día y la noche y las condiciones meteorológicas. Además, al utilizar microondas, mide la reflectividad de la escena, siendo mucho más sensible a las características del objeto relacionadas con la geometría y las propiedades eléctricas.

#### 2.2 Polarimetría SAR

Los sistemas PolSAR son sistemas SAR multidimensionales que pueden obtener más información de la escena explotando la naturaleza vectorial de las ondas electromagnéticas, empleando diferentes polarizaciones de onda para la transmisión y recepción, con el objetivo de obtener varias imágenes SAR del mismo elemento o celda de resolución. En esta sección se explican las bases de su funcionamiento.

##### 2.2.1 Dispersión Polarimétrica

Cuando la misma antena es utilizada tanto en transmisión como en recepción se considera un sistema SAR monoestático.[1].

Se define el vector del campo eléctrico con dos estados de polarización ortogonales. Tomamos el campo con polarización  $h$  como horizontal y  $v$  como vertical.

$$\begin{pmatrix} E_h^s \\ E_v^s \end{pmatrix} = \frac{\exp(-jkr)}{r} \begin{pmatrix} S_{hh} & S_{hv} \\ S_{vh} & S_{vv} \end{pmatrix} \begin{pmatrix} E_h^i \\ E_v^i \end{pmatrix} \quad (1)$$

Donde  $E$  representa el campo eléctrico,  $r$  es la distancia de la antena al objetivo.  $\underline{S}$  es la matriz de dispersión que relaciona la onda incidente y de dispersión. La convención entre ambas ondas (asumiendo el *Backward Scattering Alignment*, BSA) es  $h_s = h_i$ ,  $v_s = v_i$  y  $k_s = -k_i$

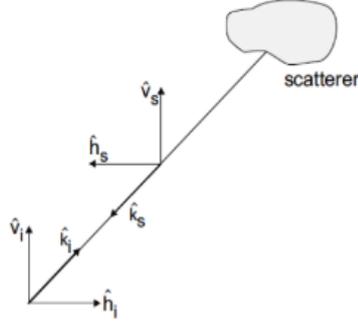


Figure 1: Eje de coordenadas de la onda incidente y dispersa. Imagen obtenida de la referencia[1]

La descomposición de Pauli[4] representa la matriz de dispersión  $\underline{\underline{S}}$  en las llamadas bases de Pauli, las cuales se definen de la siguiente forma.

$$\begin{aligned} \underline{\underline{S}}_a &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & \underline{\underline{S}}_b &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \\ \underline{\underline{S}}_c &= \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & \underline{\underline{S}}_d &= \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \end{aligned}$$

En este trabajo tenemos únicamente el caso de sistemas monoestáticos, por lo que se considera  $S_{hv} = S_{vh}$ . Con lo cual queda simplificada la descomposición como:

$$\underline{\underline{S}} = \begin{pmatrix} S_{hh} & S_{hv} \\ S_{hv} & S_{vv} \end{pmatrix} = \alpha \underline{\underline{S}}_a + \beta \underline{\underline{S}}_b + \gamma \underline{\underline{S}}_c$$

Donde:

$$\alpha = \frac{S_{hh} + S_{vv}}{\sqrt{2}} \quad \beta = \frac{S_{hh} - S_{vv}}{\sqrt{2}} \quad \gamma = \sqrt{2}S_{hv} \quad (2)$$

### 2.2.2 Representación de imágenes PolSAR

Para representar las imágenes PolSAR se puede utilizar la descomposición de Pauli para visualizar la información como si de una imagen SAR se tratase[4].

La información polarimétrica puede representarse como una combinación de intensidades  $|S_{hh}|^2$ ,  $|S_{vv}|^2$  y  $2|S_{hv}|^2$  en una única imagen RGB, donde cada una de las intensidades corresponde a uno de los canales de la siguiente manera:

$$\text{Rojo} \rightarrow |\alpha|^2 \quad \text{Verde} \rightarrow |\beta|^2 \quad \text{Azul} \rightarrow |\gamma|^2 \quad (3)$$

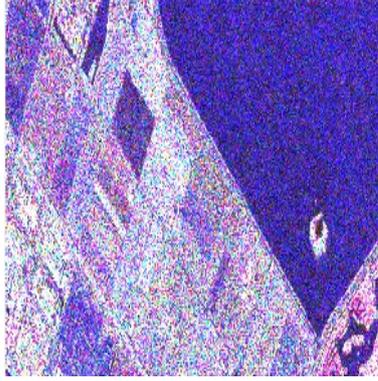


Figure 2: Representación RGB.

### 2.3 Distribución de dispersión

La distribución de dispersión asume que hay una elevada cantidad de objetos en cada celda de resolución. Una celda de resolución es una región de tamaño  $\delta_r \times \delta_r$  tal como se muestra en la figura.

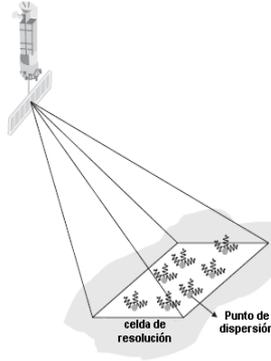


Figure 3: Representación de una celda de resolución con múltiples objetivos[1]

La onda reflejada de la escena es una combinación coherente de todas los elementos comprendidos dentro de la celda de resolución. Considerando que  $T$  es el vector transpuesto, la matriz  $S$  es una variable aleatoria y la información es extraída de sus estadísticas.  $S$  es vectorizado como el vector de dispersión  $\underline{K}_{3P}$  para simplificar esta caracterización.

$$\underline{K}_{3P} = \frac{1}{\sqrt{2}}[S_{hh} + S_{vv}, S_{hh} - S_{vv}, 2 * S_{hv}]^T \quad (4)$$

Donde se emplea la matriz de Pauli y consideramos  $S_{hv} = S_{vh}$  (caso monoestático), obteniendo una expresión más compacta.

De forma similar, mediante la matriz lexicográfica, el vector  $\underline{K}_{3L}$  de dispersión puede ser definido como:

$$\underline{K}_{3L} = [S_{hh}, \sqrt{2}S_{hv}, S_{vv}]^T \quad (5)$$

Finalmente, un sistema PolSAR mide todos los puntos de la matriz de dispersión para cada elemento de resolución. Para transmisión y recepción,  $S$  puede ser visto como 4 imágenes SAR individuales, correspondiendo con la combinación de estados de polarización para  $t_x$  y  $r_x$ . Cada

imagen sigue una distribución Gaussiana de media 0 y el vector de dispersión  $\underline{k}$  sigue una distribución Gaussiana multidimensional de media 0, identificada por la matriz de covarianza  $\underline{C}$ .

$$P_{\underline{k}}(\underline{k}) = \frac{1}{\pi^p |\underline{C}|} \exp(-\underline{k}^H \underline{C}^{-1} \underline{k}) \quad (6)$$

Donde  $p$  es la dimensión del vector de dispersión y el superíndice  $H$  es la matriz Hermitica. Esta distribución puede referirse a  $N(0, \underline{C})$ . Para el caso monoestático la matriz de covarianza  $\underline{C}$  se define como:

$$\underline{C} = E\{\underline{k}_{3L} \underline{k}_{3L}^H\} = \begin{pmatrix} E\{S_{hh} S_{hh}^*\} & \sqrt{2}E\{S_{hh} S_{hv}^*\} & E\{S_{hh} S_{vv}^*\} \\ \sqrt{2}E\{S_{hv} S_{hh}^*\} & E\{S_{hv} S_{hv}^*\} & \sqrt{2}E\{S_{hv} S_{vv}^*\} \\ E\{S_{vv} S_{hh}^*\} & \sqrt{2}E\{S_{vv} S_{hv}^*\} & E\{S_{vv} S_{vv}^*\} \end{pmatrix} \quad (7)$$

De forma equivalente, la matriz de coherencia  $\underline{T}$  se calcula de la siguiente manera.

$$\underline{T} := \langle \underline{k}_{3p} \underline{k}_{3p}^+ \rangle = \frac{1}{2} \begin{pmatrix} \langle |S_{hh} + S_{vv}|^2 \rangle & \langle (S_{hh} + S_{vv})(S_{hh} - S_{vv})^* \rangle & 2\langle (S_{hh} + S_{vv})S_{hv}^* \rangle \\ \langle (S_{hh} - S_{vv})(S_{hh} + S_{vv})^* \rangle & \langle |S_{hh} - S_{vv}|^2 \rangle & 2\langle (S_{hh} - S_{vv})S_{hv}^* \rangle \\ 2\langle S_{hv}(S_{hh} + S_{vv})^* \rangle & 2\langle S_{hv}(S_{hh} - S_{vv})^* \rangle & 4\langle |S_{hv}^*| \rangle \end{pmatrix} \quad (8)$$

## 2.4 Matrices de covarianza y coherencia de muestras

En el apartado anterior se han definido las matrices de covarianza y coherencia pero, en una situación real, estas matrices tienen que ser estimadas a partir de los datos. Por lo general, la estimación de ambas matrices se realiza mediante el promediado de las matrices de  $n$  píxeles, este proceso se denomina *multilook*. Para el cálculo de la matriz de covarianza estimada se deberá emplear  $k_{3L}$ , para el cálculo de la matriz de coherencia será  $k_{3P}$ , representamos ambas con  $k$ . La matriz calculada se representa como  $Z$ :

$$\underline{Z} = \langle \underline{k} \underline{k}^H \rangle_n = \frac{1}{n} \sum_{i=1}^n k_i k_i^H \quad (9)$$

Se puede hacer una mejor estimación incrementando la cantidad de píxeles empleados para promediar. Mientras la cantidad de píxeles empleados en el promediado no sea demasiado elevada no se perderá mucha resolución, es equivalente a aplicar un filtro paso bajo.

## 2.5 Descriptores PolSAR

Para encontrar un modo de clasificar adecuadamente los objetos de las imágenes PolSAR, es necesario el uso de descriptores. Estos descriptores deben representar de forma precisa y específica la información del nodo analizado.

### 2.5.1 Matriz de coherencia $\underline{T}$

La matriz de coherencia  $\underline{T}$  describe las propiedades de correlación entre las características físicas extraídas de la información polimétrica, por ese motivo se han utilizado los valores de esta matriz como descriptores, ya que la matriz  $\underline{T}$ , junto con la matriz  $\underline{C}$  son la mayor fuente de información posible.

Sólo se tendrán en cuenta los valores de la parte superior de la diagonal y la propia diagonal ya que allí tenemos toda la información polimétrica.

### 2.5.2 Descomposición de Hoekman

Dirk H. Hoekman propone una transformación reversible de la matriz de covarianza en intensidades de dispersión con el objetivo de describir toda la información polarimétrica con una representación matemática alternativa[6].

#### Transformación

Todas las propiedades polarimétricas pueden describirse mediante la matriz de covarianza definida anteriormente 7.

Estas propiedades pueden encontrarse en tres números reales en la diagonal y en seis valores complejos alrededor de la diagonal, de los cuales únicamente tres nos aportan información ya que es una matriz hermítica. Otra forma de representar toda la información polarimétrica, relacionada con la percepción física, es empleando tres valores de intensidad, tres fases de polarización distintas y tres valores coherentes de magnitud (representando  $\rho$  valores complejos).

$$\underline{\underline{C}} = 4\pi \begin{pmatrix} \sqrt{\sigma_{hh}^0} & 0 & 0 \\ 0 & \sqrt{\sigma_{hv}^0} & 0 \\ 0 & 0 & \sqrt{\sigma_{vv}^0} \end{pmatrix} \begin{pmatrix} 1 & \rho_{hhhv} & \rho_{hhvv} \\ \rho_{hhhv}^* & 1 & \rho_{hvvv} \\ \rho_{hhvv}^* & \rho_{hvvv}^* & 1 \end{pmatrix} \begin{pmatrix} \sqrt{\sigma_{hh}^0} & 0 & 0 \\ 0 & \sqrt{\sigma_{hv}^0} & 0 \\ 0 & 0 & \sqrt{\sigma_{vv}^0} \end{pmatrix} \quad (10)$$

También es posible describir toda la información polarimétrica empleando nueve intensidades.

$$\underline{\underline{C}} = \begin{pmatrix} \langle S_{hh}S_{hh}^* \rangle \\ \langle S_{vv}S_{vv}^* \rangle \\ \langle S_{hv}S_{hv}^* \rangle \\ Re[\langle S_{hh}S_{vv}^* \rangle] \\ Im[\langle S_{hh}S_{vv}^* \rangle] \\ Re[\langle S_{hh}S_{hv}^* \rangle] \\ Im[\langle S_{hh}S_{hv}^* \rangle] \\ Re[\langle S_{hv}S_{vv}^* \rangle] \\ Im[\langle S_{hv}S_{vv}^* \rangle] \end{pmatrix} = \underline{\underline{B}} \begin{pmatrix} \sigma_{hh}^0 \\ \sigma_{vv}^0 \\ \sigma_{++45}^0 \\ \sigma_{--45}^0 \\ \sigma_{II}^0 \\ \sigma_{rr}^0 \\ \sigma_{h+45}^0 \\ \sigma_{hI}^0 \\ \sigma_{+45I}^0 \end{pmatrix}, \text{ con } \underline{\underline{B}} = \begin{pmatrix} 1 & 0 & 0 & | & 0 & 0 & 0 & | & 0 & 0 & 0 \\ 0 & 1 & 0 & | & 0 & 0 & 0 & | & 0 & 0 & 0 \\ -\frac{1}{4} & -\frac{1}{4} & \frac{1}{4} & | & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & | & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & | & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & | & 0 & 0 & 0 \\ \frac{1}{4} & \frac{1}{4} & \frac{3}{4} & | & -\frac{1}{4} & \frac{3}{4} & -\frac{1}{4} & | & 0 & 0 & -2 \\ -\frac{3}{8} & \frac{1}{8} & -\frac{1}{8} & | & -\frac{1}{8} & -\frac{1}{8} & -\frac{1}{8} & | & 1 & 0 & 0 \\ \frac{3}{8} & -\frac{1}{8} & \frac{1}{8} & | & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & | & 0 & -1 & 0 \\ \frac{3}{8} & -\frac{1}{8} & \frac{1}{8} & | & -\frac{3}{8} & \frac{1}{8} & -\frac{1}{8} & | & -1 & 0 & 0 \\ -\frac{3}{8} & \frac{1}{8} & -\frac{1}{8} & | & -\frac{1}{8} & -\frac{1}{8} & \frac{3}{8} & | & 0 & 1 & 0 \end{pmatrix} \quad (11)$$

El desarrollo de este método puede verse en el apéndice de la referencia.

Algunos campos pueden presentar un comportamiento polarimétrico asimutal asimétrico. Por ejemplo, un campo de patata con orientación de 45° con respecto a la dirección de vuelo difiere de un campo idéntico con una orientación de 135° en su firma polarimétrica. La información "extra" sobre la orientación puede perturbar el proceso de clasificación. Una solución al problema es descartar los valores que presentan dicho problema. En tal caso la matriz de covarianza queda de esta manera.

$$\underline{\underline{C}} = \begin{pmatrix} \langle S_{hh}S_{hh}^* \rangle & 0 & \langle S_{hh}S_{vv}^* \rangle \\ 0 & \langle S_{hv}S_{hv}^* \rangle & 0 \\ \langle S_{vv}S_{hh}^* \rangle & 0 & \langle S_{vv}S_{vv}^* \rangle \end{pmatrix} \quad (12)$$

$$\underline{\underline{C}} = \begin{pmatrix} \langle S_{hh}S_{hh}^* \rangle \\ \langle S_{vv}S_{vv}^* \rangle \\ \langle S_{hv}S_{hv}^* \rangle \\ Re[\langle S_{hh}S_{vv}^* \rangle] \\ Im[\langle S_{hh}S_{vv}^* \rangle] \end{pmatrix} = \underline{\underline{B}} \begin{pmatrix} \sigma_{hh}^0 \\ \sigma_{vv}^0 \\ \sigma_{hv}^0 \\ \sigma_{+-45}^0 \\ \sigma_{+45I}^0 + \sigma_{-45I}^0 \end{pmatrix}, \text{ con } \underline{\underline{B}} = \begin{pmatrix} 1 & 0 & 0 & | & 0 & 0 \\ 0 & 1 & 0 & | & 0 & 0 \\ 0 & 0 & 1 & | & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & | & -2 & 0 \\ \frac{1}{2} & \frac{1}{2} & 1 & | & 0 & -1 \end{pmatrix} \quad (13)$$

Los parámetros  $\sigma$  son los que serán utilizados como descriptores. Donde  $\sigma_{hh}^0, \sigma_{vv}^0$  y  $\sigma_{hv}^0$  representan la potencia de  $S_{hh}, S_{vv}$  y  $S_{hv}$  a  $\sigma_{+-45}^0$  es la potencia obtenida en base de polarización lineal,  $\sigma_{+45I}^0 + \sigma_{-45I}^0$  es la potencias obtenida en base a polarización circular.

Analizando los parámetros  $\sigma_{hv}^o$  y  $\sigma_{+-45}^o$  y comparándolos con los valores de la diagonal de la matriz de coherencia  $\underline{T}$  Se llega a la conclusión que  $\langle |S_{hh} - S_{vv}|^2$  y  $\sigma_{+-45}^o$  representan la misma información. Lo mismo ocurre con  $4\langle |S_{hv}^*| \rangle$  y  $\sigma_{hv}^o$ . Por lo que se decide descartar  $\sigma_{hv}^o$  y  $\sigma_{+-45}^o$ .

### 2.5.3 Descomposición en autovalores, H, A y alpha

#### Autovalores

Con el calculo de los autovalores de la matriz de coherencia  $\underline{T}$ , podemos obtener una descomposición que representa las propiedades físicas características del objeto analizado[4]. Conforme el teorema de la descomposición, la matriz  $\underline{T}$  puede descomponerse como:  $\underline{T} = \underline{U}\underline{\Sigma}\underline{U}^{-1}$  Donde  $\underline{\Sigma}$  es una matriz diagonal formada por los autovalores de  $\underline{T}$ .

$$\underline{\Sigma} = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix} \quad (14)$$

Con  $\infty > \lambda_1 > \lambda_2 > \lambda_3 > 0$ . Empleando los autovalores podemos calcular el Span, que es la energía total de la región. Se define como:  $Span = \lambda_1 + \lambda_2 + \lambda_3$

#### Entropía

La entropía define el grado de heterogeneidad de los autovalores  $\underline{T}$ .

$$H = \sum_{i=1}^3 p_i \log_3(p_i) \quad \text{Donde} \quad p_i = \frac{\lambda_i}{\sum_{k=1}^3 \lambda_k} \quad (15)$$

Donde se considera  $p_i$  como la importancia relativa del autovalor en proporción al total de energía del elemento.

#### Anisotropía

Evalúa la importancia del segundo y tercer valores propio.  $A = \frac{\lambda_2 - \lambda_3}{\lambda_2 + \lambda_3}$

#### Ángulo alpha

El parámetro alpha se puede relacionar con las propiedades físicas del proceso de dispersión.

$$\alpha = \sum_{i=1}^3 p_i \alpha_i \quad (16)$$

$\alpha \rightarrow 0$  : La dispersión corresponde a la dispersión de un solo rebote producida por una superficie.

$\alpha \rightarrow \frac{\pi}{4}$  : El mecanismo de dispersión corresponde a la dispersión de volumen.

$\alpha \rightarrow \frac{\pi}{2}$  : El mecanismo de dispersión se debe a la dispersión de doble rebote.

## 2.6 Binary Partition Tree

En esta sección se describe la estructura de *Binary Partition Tree*.

### 2.6.1 Representación BPT

BPT puede considerarse como una representación multiescala basada en regiones. En esencia, su funcionamiento consiste en conectar áreas conexas de la imagen en función de su similitud.

La estructura jerárquica es la siguiente: las hojas representan regiones que corresponden a los píxeles originales de la imagen. Los nodos superiores se crean a partir de la fusión de distintas hojas. La combinación de nodos continua a lo largo de toda la estructura hasta llegar al nodo raíz, el cual representa la imagen al completo [8].

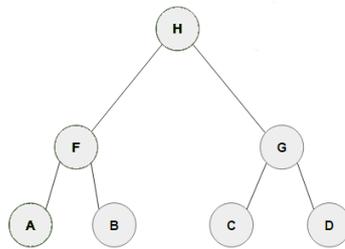


Figure 4: Representación del BPT

En la imagen se puede ver un ejemplo visual de la estructura y composición del BPT. Los nodos A, B, C y D son hojas. El nodo F está formado por las hojas A y B. De la misma manera, G está compuesto por C y D. El nodo H está compuesto por los nodos F y G, por lo que en conjunto está formado por las hojas A, B, C y D, representando toda la imagen.

#### Ejemplo visual

Para hacer un ejemplo simple del funcionamiento del BPT se va a utilizar esta imagen 4 píxeles. Donde la figura anterior es el diagrama del BPT asociado a la imagen.

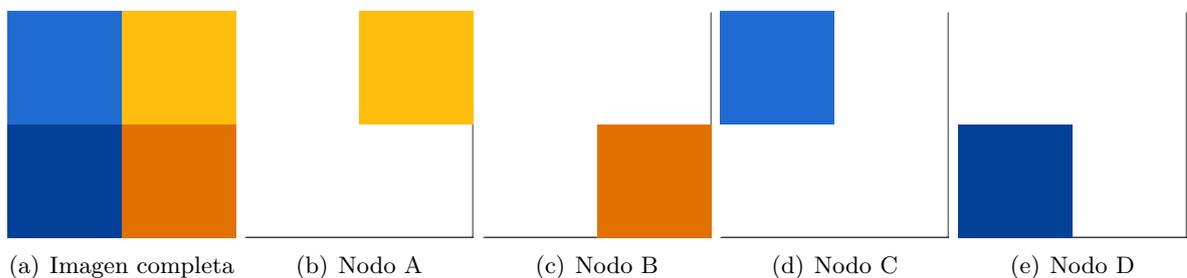


Figure 5: Cada una de las hojas representa un píxel

Visualizamos los nodos F, G y H.

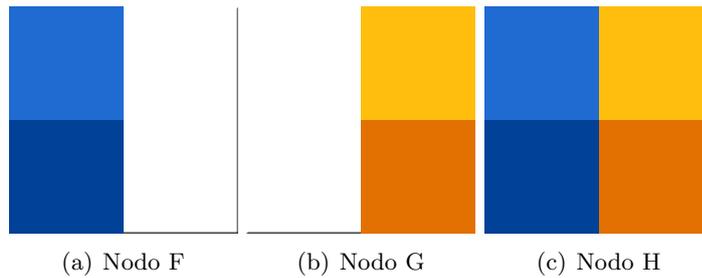


Figure 6: Los nodos representan la información de las hojas que los forman.

### 2.6.2 Conectividad de la imagen

En el momento de generar el BPT tenemos dos opciones en cuanto a la conectividad. Escogiendo una conectividad cuatro, se observan los valores superiores y laterales que rodean al píxel, pero no se toman en cuenta los valores diagonales. Esta es la opción que tiene menor coste de cálculo. Si escogemos una conectividad ocho observamos todos los valores circundantes al píxel y se pueden representar estructuras físicas en cualquier dirección. En este proyecto utilizamos la conectividad 8.

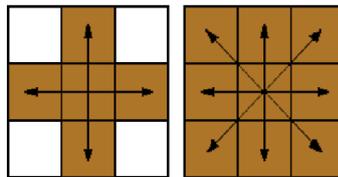


Figure 7: A la izquierda ejemplo de conectividad cuatro. A la derecha ejemplo de conectividad ocho

### Modelo de región

El modelo es la matriz de covarianza promediada de todos los píxeles de la región. Entendiendo como región todos los píxeles pertenecientes al nodo.

### 2.6.3 Medidas de similitud

La similitud entre regiones define la secuencia de fusión del árbol. Las primeras regiones en fusionarse serán aquellas más similares entre si. Hay dos tipos de medidas: Aquellas que emplean todos los valores de la matriz de covarianza y aquellas que únicamente emplean los valores de la diagonal[1]. En este proyecto se utilizan todos los valores de la matriz.

## 2.7 Algoritmos de clasificación supervisados

Los algoritmos de clasificación supervisados son aquellos que mediante un conjunto de datos etiquetados como pertenecientes a dos o más clases, son capaces de crear un modelo de clasificación. Un modelo de clasificación es un algoritmo capaz de identificar conjuntos de datos como perteneciente a una clases mediante la identificación de sus características. [9]

### 2.7.1 Support Vector Machine (SVM)

SVM calcula un hiperplano para separar de forma óptima a los puntos de una clase de la de otra. Se busca encontrar el plano que maximiza la separación entre clases, de esta forma, los elementos a un lado del hiperplano son clasificados como pertenecientes a una clase y los que estén al otro lado se consideran de una clase distinta. La búsqueda de la separación óptima es la característica principal de SVM.

Es un sistema muy robusto al sobreentrenamiento. Puede trabajar con muestras que tengan un gran número de descriptores, aun así es muy sensible a la selección de las características introducidas como parámetro de entrada. La velocidad de entrenamiento del modelo es muy baja. Una de sus principales ventajas de SVM es la posibilidad de utilizar kernels. Un kernel es una medida de similitud empleada por el sistema de clasificación. Con el kernel apropiado puede trabajar incluso con datos que no son linealmente separables. En este proyecto se ha empleado el kernel Gaussiano (o Radial).

#### Kernel Gaussiano

*Radial basis function* o RBF [10], es un kernel comúnmente utilizado en varios sistemas de *machine learning*, especialmente en SVM, donde dos muestras  $x$  y  $x'$  representan las características de entrada al sistema.  $K(x, x') = \exp(-\frac{\|x-x'\|^2}{2\sigma^2})$ . Puede interpretarse  $\|x-x'\|^2$  como la distancia euclídea.

La mejor forma de entender este kernel es verlo como una distribución Gaussiana.

El kernel Gaussiano es una función exponencial decreciente, con el valor máximo en el vector de soporte decayendo uniformemente en todas las direcciones a su alrededor.

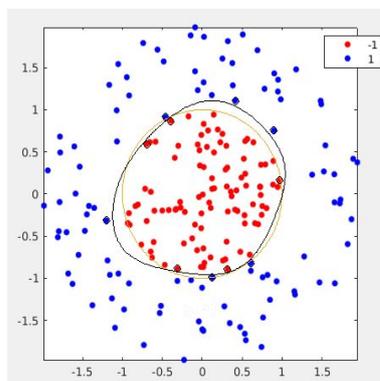


Figure 8: Separación entre clases. Las fronteras representadas en amarillo y negro son ejemplos de propuestas del sistema mientras busca la separación óptima, se puede ver que la frontera entre puntos de la dos clases se va modificando.

La figura muestra como el sistema busca la separación óptima entre las dos clases definiendo la frontera entre ambas. Se realizan diferentes iteraciones buscando la frontera que otorgue la mayor separación entre muestras. <https://www.quora.com/What-is-the-intuition-behind-Gaussian-kernel-in-SVM>

### 2.7.2 Árbol de decisión

Un árbol de decisión tiene como objetivo crear un modelo que predice el valor de una clase de destino en función de diversas variables de entrada. Se crea una estructura empleando el parámetro que mejor pueda separar un mayor número de elementos en cada nodo. Cada nodo interior corresponde a una de las variables de entrada. En estas estructuras de árbol, las hojas representan etiquetas de clase y las ramas representan las conjunciones de características que conducen a esas etiquetas de clase.

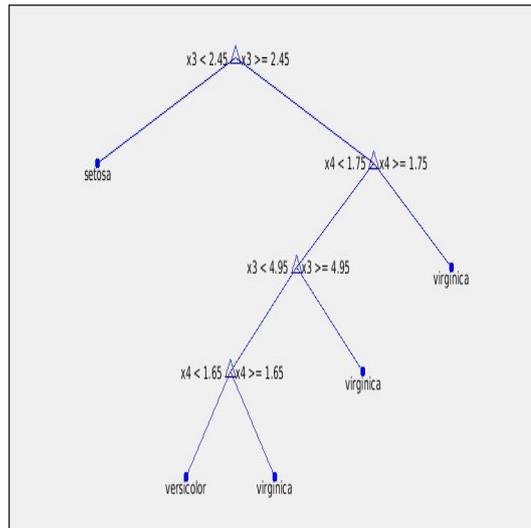


Figure 9: Representación de la estructura de un árbol. La base de datos utilizada para representar este árbol es *fisheriris*, extraída del repositorio de Matlab

Puede clasificar datos no lineales. No se ven afectado por tratar con parámetros redundantes, son robustos al ruido y se crea el modelo en poco tiempo. Por contra, puede sobreentrenarse fácilmente, no puede reconocer interacciones complejas entre parámetros.

## 2.8 Evaluación de los resultados, Fscore

Precisión y exhaustividad son dos parámetros empleados para medir el rendimiento de los sistemas de búsqueda y recuperación de información y reconocimiento de patrones.

Precisión: es un parámetro que expresa la proporción entre la cantidad de muestras clasificadas correctamente de entre todas las clasificadas.

Exhaustividad: este parámetro expresa la proporción de elementos clasificados correctamente, comparado con el total de los elementos existentes.

Fscore es un parámetro ponderado entre la precisión y la exhaustividad empleado para medir el resultado de la clasificación con un único valor.

$$F_{score} = 2 \frac{\text{precision} \times \text{exhaustividad}}{\text{precision} + \text{exhaustividad}} \quad (17)$$

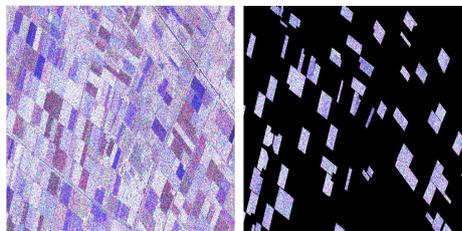
Para medir los resultados obtenidos de la clasificación para cada modelo se calcula el Fscore.

## 2.9 Datos de entrenamiento y test

Para entrenar los modelos de clasificación y probar los modelos creados se emplean dos tipos de datos distintos. Se ha seleccionado una región de la base de datos de tamaño 1000 x 1000 píxeles para entrenar el modelo. La imagen empleada fue tomada en la campaña ESA AgriSAR, y pertenece a Flevoland, en los países bajos. Se han tomado diferentes regiones de tamaño 250 x 250 píxeles para comprobar su eficacia. Para realizar las pruebas se ha decidido reducir el problema de 22 clases a 2 para simplificarlo. Una de estas clases corresponde a los píxeles etiquetado como cultivos de patatas y la otra clase corresponde a todos los píxeles que no sean de patata.

### 2.9.1 Datos de entrenamiento

La región de 1000 x 1000 píxeles ha sido seleccionada de este tamaño para tener datos suficientes para el entrenamiento. La región ha sido escogida por contar con elementos pertenecientes a las 22 clases originales.



(a) Imagen original      (b) Clase patata

### 2.9.2 Datos de test

Para probar el modelo entrenado he seleccionado diferentes regiones de la imagen original del *dataset*. Estas imágenes son de tamaño 250 x 250. Han sido seleccionadas por contar con un porcentaje elevado de píxeles de patata. Las regiones usadas para el test no pertenecen a la región empleada para crear el modelo.

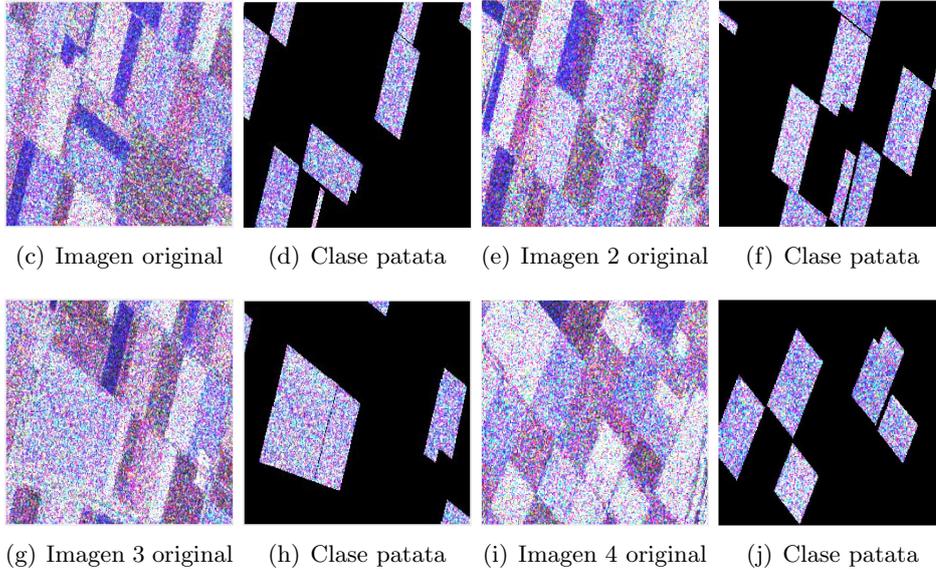


Figure 10: Imágenes de test.

## 2.10 Normalización

Es necesario normalizar los descriptores para poder crear modelos y clasificar los datos. Los datos tendrán media 0 y varianza 1, de este modo todos los descriptores tendrán aproximadamente el mismo margen dinámico.

$$X' = \frac{X - \mu}{\sigma} \quad (18)$$

Donde  $X$  representa el conjunto de valores del descriptor en la base de datos,  $\sigma$  la varianza del conjunto correspondiente al descriptor,  $\mu$  su valor medio y  $X'$  representa los valores del descriptor normalizados. Se normalizan los datos de test con los mismos parámetros empleados para normalizar los datos usados para crear el modelo.

## 2.11 Estado del arte clasificación PolSAR

En este apartado se habla de los diferentes tipos de trabajos previos sobre clasificación de imagen PolSAR.

### **A New Polarimetric Classification Approach Evaluated for Agricultural Crops[6]**

Son varios los autores en que emplean descomposiciones polarimétricas para definir descriptores para la clasificación. La descomposición de Hoekman se ha empleado para clasificar campos de cultivo de 14 tipos distintos. Se emplea la descomposición como descriptores para los píxeles y se procede a su clasificación. En el documento de referencia no se especifica el método empleado para clasificar los píxeles.

### **A Novel Polarimetric-Texture-Structure Descriptor for High-Resolution PolSAR Image Classification[11]**

El elemento más destacable de este documento es que emplea como herramienta para la

clasificación el BPT. Para la clasificación se evalúan los 3 nodos jerárquicamente superiores al elemento que se está observando y se añaden sus descriptores a los del píxel. Como descriptores se utilizan la descomposición de Freeman y Durden, Span, la elongación del nodo, compactación de la elipse, relación de escala, área normalizada y la homogeneidad de la región. Para la clasificación se emplea SVM con *intersection kernel* (HIK). Se clasifican 6 clases distintas de terrenos. Los elementos clasificados son de tamaño 100x100 píxeles.

## Capítulo 3

### 3 Algoritmo de clasificación con BPT

#### 3.1 Idea principal

El objetivo de este proyecto es mejorar la clasificación basada en píxeles mediante la clasificación de los nodos pertenecientes a la rama de dicho píxel. Para esto es necesario realizar dos pasos. El primero, consiste en clasificar todos los nodos del BPT y obtener la probabilidad a posteriori de que pertenezcan a las clases. El segundo paso es clasificar la evolución de la probabilidad a posteriori a lo largo de la rama del píxel. Por lo tanto es necesario realizar dos clasificaciones.

#### 3.2 BPT para mejorar la clasificación

Empleando la rama del BPT a la que pertenece el píxel podemos obtener información que nos permita clasificarlo mejor. Una rama es el conjunto de nodos que unen una hoja con el nodo raíz. A un nodo jerárquicamente superior a otro de la misma rama se le denomina nodo padre. Las ramas de un BPT pueden tener diferentes longitudes. Hay tantas ramas como píxeles en la imagen, las ramas pueden tener elementos en común entre ellas pero todas son distintas.

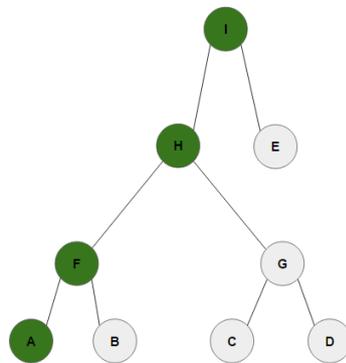


Figure 11: Representación de la rama de la hoja A, de la cual forman parte los nodos A, F, H y I. El nodo F es padre del nodo A, así mismo los nodos F y G son hijos de H.

El primer paso consiste en realizar una clasificación de los nodos de cada rama, en esta clasificación se definirá la clase a la que pertenece el nodo y la probabilidad a posteriori de que pertenezca a las clases. Se emplearán las probabilidad a posteriori de la rama para crear un modelo de clasificación y así, finalmente, decidir si una hoja pertenece a una clase o no.

##### 3.2.1 Clasificación de los nodos

En esta sección se explica como se realiza la clasificación de los nodos del árbol.

### 3.2.2 Descriptores

Se calculan los descriptores de Hoekman, la matriz de coherencia  $\underline{T}$ , autovalores, Entropía, Anisotropía y ángulo alpha para todos elementos del BPT. Para calcular los descriptores de los nodos, primero se calculan los descriptores para los píxeles de la imagen, una vez calculados, se promedian los descriptores de todos los píxeles que pertenecen al nodo, excepto para el caso de la Entropía, la Anisotropía y ángulo alpha, que se calculan a partir de  $\underline{T}$ .

Para realizar la clasificación se emplea el clasificador SVM con kernel Gaussiano.

## 3.3 Ramas del BPT

### 3.3.1 Representación de las ramas

Para poder visualizar la información contenida en el BPT se realiza el siguiente tipo de visualización, en el que representamos todo el BPT en una única imagen. El eje  $x$  corresponde a los píxeles de la imagen. El eje  $y$  corresponde a los nodos de la rama del pixel. Colocando la raíz del BPT en la parte inferior del gráfico, en la parte superior se colocan las hojas. Hay ramas de diferentes longitudes, en las ramas más cortas a los valores inexistentes se les da valor 0, la imagen del siguiente apartado es un ejemplo.

### 3.3.2 Mezcla de clases

Por la estructura del BPT, habrá nodos de la rama en los que coexistirán píxeles que pertenecerán a diferentes clases.

En el siguiente gráfico se representan todas las ramas del BPT. Mostramos el porcentaje de píxeles de la clase patata que tiene cada nodo.

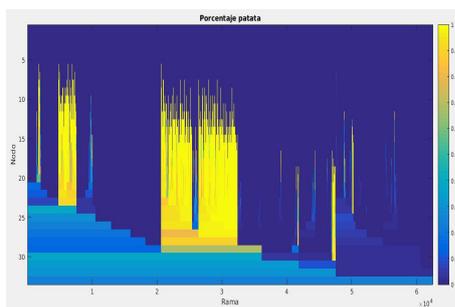


Figure 12: Representación del porcentaje de patata en cada nodo. El color amarillo representa 100% de patata.

Puede verse que si la hoja es de clase patata, sus nodos padre más cercanos también lo son. La mezcla de clases no se produce hasta llegar a nodos cercanos a la raíz. Por lo tanto concluimos que tenemos ramas largas que agrupan una gran cantidad de píxeles de la misma clase.

### 3.4 Análisis de los nodos

En este apartado se habla de como son tratados los nodos del árbol y que información podemos concluir gracias a ellos.

#### 3.4.1 Identificación de la clase del nodo

Aquellos nodos cuyos píxeles pertenecen a una única clase se etiquetan como pertenecientes de la misma, pero por la estructura del BPT, en algunos nodos coincidirán píxeles de distintas clases. Esto presenta la necesidad de definir cuando podemos considerar a un nodo como miembro de la clase y cuando no, de cara a hacer un entrenamiento que incluya nodos que tengan hojas de distintas clases y se mejore la clasificación. Se empieza observando cuantos elementos de cada clase hay en el nodo y otorgarle al nodo la clase del elemento dominante. Se plantea lo siguiente: hay un limite superior y otro inferior, el limite superior define el porcentaje mínimo de elementos de una clase que ha de tener el nodo para que sea considerado como tal. El limite inferior nos indica a partir de que porcentaje de elementos se ha clasificar al nodo como fuera de la clase. Los nodos que estén entre esos dos limites no se incluirán en el entrenamiento.

Definimos un parámetro al que llamaremos pureza, que indicara que porcentaje de elementos de la clase dominante hay en cada nodo, por ejemplo, pureza 100% indica que todos los elementos del nodo son de la misma clase y pureza 90% indica que sólo 9 de cada 10 elementos del nodo son de la clase dominante. Se realiza una prueba inicial en que se separan los nodos a intervalos según su pureza. Todos los nodos con valores de pureza comprendidos entre el 90% y 100% se clasifican y se evalúa el Fscore de la clasificación. Luego aquellos entre 80% y 90%, se vuelve a evaluar la clasificación y así sucesivamente. Los resultados son los siguientes:

<b>Porcentaje</b>	<b>100%</b>	<b>99%-90%</b>	<b>90%-80%</b>	<b>80%-70%</b>	<b>70%-60%</b>	<b>60%-50%</b>
<b>Fscore</b>	0.886	0.885	0.704	0.568	0.435	0.322

Table 1: Fscore por porcentaje de elementos de la misma clase.

El Fscore de todos los nodos es de 0.834. Tomando estos resultados se puede ver como el sistema clasifica con el mismo ratio de éxito elementos con una pureza de 90% o superior.

Se realizan unos entrenamientos donde se considera que si la pureza es 90% o superior pertenece a la clase. Por lo tanto definimos que el límite superior es 90% y se prueban diversos limites inferiores.

<b>Limite inferior</b>	<b>90%-80%</b>	<b>80%-70%</b>	<b>70%-60%</b>	<b>60%-50%</b>
<b>Fscore</b>	0.825	0.834	0.834	0.834

Table 2: Fscore considerando como límite superior 90%.

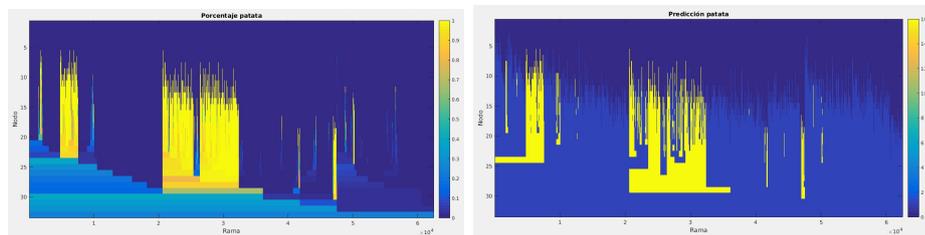
Analizando los resultados vemos que si consideramos que los nodos con pureza 80% no son de la clase empeora la clasificación. Estos nodos aun son demasiado similares a la clase como para no considerarlos de la misma. Se hacen entrenamientos donde se considera que el límite superior es 80% y se prueban diversos límites inferiores.

Limite inferior	80%-70%	70%-60%	60%-50%
Fscore	0.835	0.835	0.835

Table 3: Fscore considerando como límite superior 80%.

Los resultados de este segundo entrenamientos son ligeramente superiores al del anterior. No hay diferencia apreciable entre el porcentaje de elementos seleccionados como no pertenecientes a la clase. Para el modelo final se emplea el entrenamiento considerando a nodos con porcentaje de 80% de la clase como pertenecientes a la misma y aquellos con un porcentaje inferior al 50% como fuera de la clase. El resto no se tienen en cuenta en el entrenamiento.

### 3.4.2 Representación de la clasificación de los nodos



(a) Porcentaje de clase patata (b) Clasificación de los nodos. El amarillo representa la clase patata, el azul claro la clase no patata, el azul oscuro valores inexistentes en la rama.

Figure 13: Clasificación comparada con el porcentaje de patata del nodo.

En las gráficas puede observarse lo que ya habíamos visto en la tabla[1], los nodos con porcentaje de 100% de clase patata y los del 90% tienen aproximadamente el mismo Fscore.

Se ha evaluado cuántos elementos hay con distintos porcentajes de mezcla en el árbol. El 50% del total son hojas, por tanto sólo pertenecen a una clase. El 28.9% son nodos donde se unen diversa cantidad de píxeles de la misma clase. El 31.1% restante corresponde a los porcentajes inferiores al 100% de pertenencia a la misma clase. Eso implica que algo más del 78.9% de los nodos del son clasificados con un Fscore cercano a 0.835. Al mismo tiempo estos nodos son los más cercanos a la hoja, por lo que podemos concluir que los nodos cercanos son los más útiles para la clasificación.

La probabilidad a posteriori de cada nodo de pertenecer a la clase patata es la siguiente:

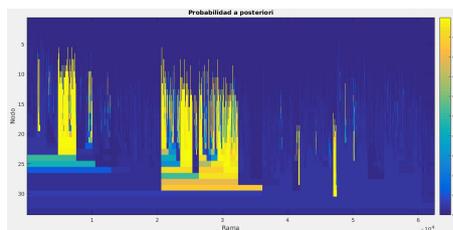


Figure 14: Representación de la probabilidad a posteriori.

La probabilidad de los nodos con porcentajes altos de patata es cercana a 1, podemos utilizar

este parámetro como indicador de que a clase pertenece el nodo y con que probabilidad de que esto sea cierto. Un píxel que se haya identificado mal, puede tener nodos en la rama que se clasifiquen correctamente y utilizarlos para realizar una segunda clasificación donde se mejore el resultado.

### 3.5 Uso de la probabilidad a posteriori para la clasificación

Se prueban diferentes técnicas para clasificar las hojas empleando la clasificación de los nodos que conforman sus ramas empleando modelos de clasificación SVM y Arboles de decisión.

#### 3.5.1 Uso de los $N$ primeros componentes

Se plantea un método en el que se emplearan los  $N$  primeros componentes de cada rama para la clasificación, ya que sabemos que los primeros nodos de la rama son los que tienen más potencial para ser útiles para la clasificación, mientras que los últimos, debido a que mezclan píxeles de diferentes clases, pierden su utilidad.

El propósito de emplear los  $N$  primeros componentes es aprovechar que la información útil para la clasificación estará en los primeros valores de la rama, antes de que empiece la mezcla de clases. El problema de este método es que cada rama es de un tamaño distinto y por lo tanto la información representada en cada nodo también.

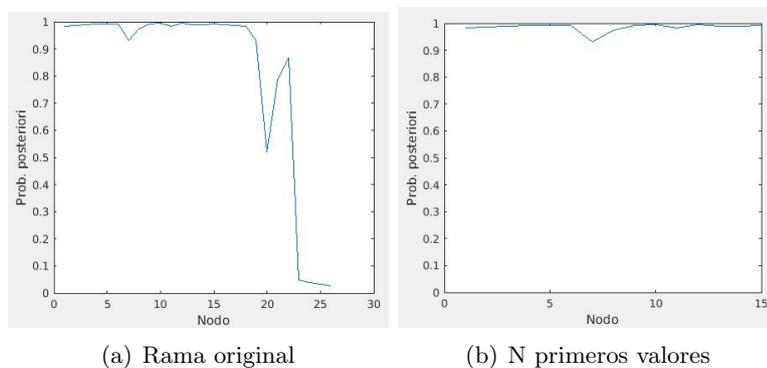


Figure 15: A la izquierda la probabilidad a posteriori de la rama. A la derecha la representación de los  $N$  primeros valores con  $N=15$ .

#### Valor de $N$

Para encontrar la cantidad adecuada de parámetros se entrena un modelo de árbol de clasificación en el que se emplean los 40 primeros y se evalúa la importancia del descriptor. Se estima la importancia del descriptor para el árbol sumando los cambios en el riesgo debido a las divisiones en cada descriptor y dividiendo la suma por el número de nodos de rama. Cuando mayor sea el valor del error asociado a cada descriptor, más importante es. En caso de valer 0, indica que el modelo no lo utiliza y por lo tanto se considera que es información inútil.

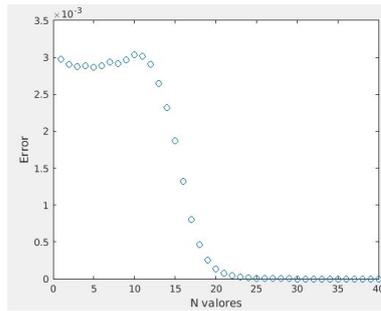


Figure 16: Representación de la importancia de los descriptores.

A la vista de los resultados, a partir del nodo número 13 la importancia decae de forma drástica. Al llegar al nodo 20 no parece que haya ninguno más que sea relevante.

### 3.5.2 Probabilidad condicionada en función del área

La probabilidad a posteriori de los nodos de una rama varía en función de la cantidad y el tipo de elementos añadidos de un nodo a otro. Con el método de N primeros componentes no conocemos información del nodo aparte de su posición respecto a la hoja. Un nodo que ocupe una posición en una rama puede ser completamente distinto en tamaño al nodo que ocupe la misma posición. Con lo cual no tenemos información útil que nos indique la importancia del parámetro. Con este método se busca representar la probabilidad condicionada en función del área de la rama haciendo una estimación empleando los valores conocidos para obtener una representación uniforme. Este método a diferencia de N primeros componentes representa la rama de un modo que la longitud original de esta no es relevante.

#### Área del nodo

Como resultado de la fusión de nodos, cada vez se obtienen nodos de mayor tamaño al estar uniendo elementos cada vez más grandes. Observando el incremento del área de las ramas del BPT puede verse una evolución proporcional de una rama a otra. En la siguiente imagen puede analizarse el incremento de área en las ramas. Esta imagen ha sido representada en función del logaritmo en base 10 para que se pueda apreciar la evolución, (dado que los valores originales están comprendidos entre 1 y 62500 los más pequeños no se apreciaban).

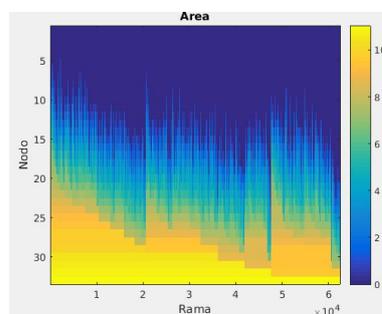


Figure 17: Representación del incremento del área en logaritmo base 10.

#### Definición de intervalos

La cantidad de elementos que podemos introducir al clasificador es limitada, no es posible introducir un parámetro para cada valor de área. Deben seleccionarse valores que representen la rama lo mejor posible. Teniendo en cuenta el incremento del área, podemos definir intervalos en función del tamaño del nodo, en los que se asignara al descriptor la probabilidad condicionada de pertenecer a la clase. En otras palabras, se definirán  $N$  intervalos entre 1 y un tamaño de área máximo. Una vez definidos, se analizaran las ramas y en función del área del nodo, se le asignara el valor de la probabilidad condicionada de pertenecer a la clase al intervalo correspondiente. Para definir los intervalos utilizaremos el algoritmo de Max-Lloyd[12].

Es de esperar que la mayoría de las ramas no tendrán un nodo con el área correspondiente a todos los intervalos definidos, en estos casos se tomaran los dos valores conocidos que comprendan los intervalos vacíos y se interpolara para asignarles un valor. En caso de no haber dos valores conocidos entre esos intervalos el valor de esos descriptores sera -1.

Se ha decidido que el tamaño máximo de área a tener en cuenta sea 4000 pues es el doble de lo que miden la mayor parte de los campos de cultivo. De esta manera sabremos que estamos incluyendo el campo completo en el análisis.

### Cantidad de intervalos

Para encontrar el número optimo de intervalos se utiliza el algoritmo de Max-Lloyd y se hacen diferentes particiones de los datos, cada partición divide los datos en una cantidad distinta. Se mide el error cuadrático medio del sistema cada vez. Como datos para crear el modelo se han empleado todos los valores de los datos de entrenamiento con área menor a 4000.

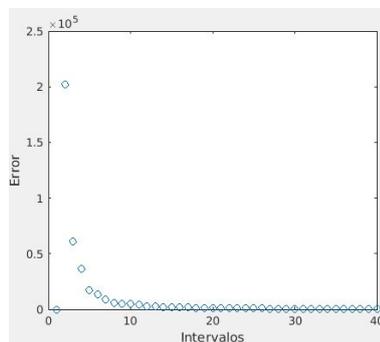


Figure 18: Error por cantidad de intervalos.

El error tiende a 0 hasta alcanzarlo al llegar a los 14 intervalos. El sistema entrenado en este trabajo utiliza 14 intervalos definidos por el algoritmo, los cuales son los siguientes: 0-115, 116-345, 346-633, 634-917, 918-1245, 1246-1580, 1581-1841, 1842-2164, 2165-2470, 2471-2651, 2652-2877, 2878-3163, 3164-3450,3451-3742.

A estos intervalos se les añadió otros para representar de forma más precisa la información en los nodos más pequeños. Además, la probabilidad a posteriori del píxel en cuestión siempre sera añadida como primer parámetro. Por lo que finalmente queda de esta manera: 1, 11-30, 31-80, 81-115, 116-345, 346-633, 634-917, 918-1245, 1246-1580, 1581-1841, 1842-2164, 2165-2470, 2471-2651, 2652-2877, 2878-3163, 3164-3450,3451-3742.

### Ejemplo

Tomamos un píxel de la clase patata y analizamos su rama, puede verse representada en

la figura inferior. Esta la rama tiene una longitud de 23 nodos. El área de dichos nodos es la siguiente: 1, 2, 3, 10, 18, 22, 75, 129, 280, 430, 670, 1451, 2799, 7568, 10453, 12930, 15880, 18161, 20579, 36081, 41835, 47548, 62500. Establecemos 4000 como área máxima a tener en cuenta, con lo cual sólo tenemos utilizaremos: 1, 2, 3, 10, 18, 22, 75, 129, 280, 430, 670, 1451, 2799. Varias de las muestras coinciden en el mismo intervalo de área, por lo que finalmente sólo emplearemos los nodos de área: 1, 22, 75, 129, 430, 670, 1451, 2799.

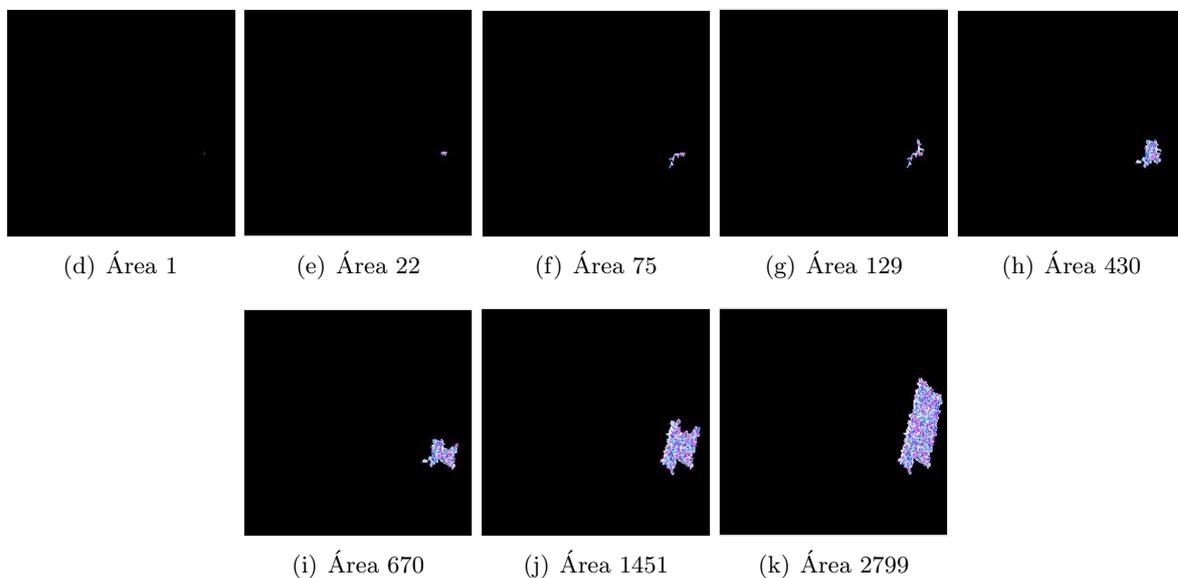
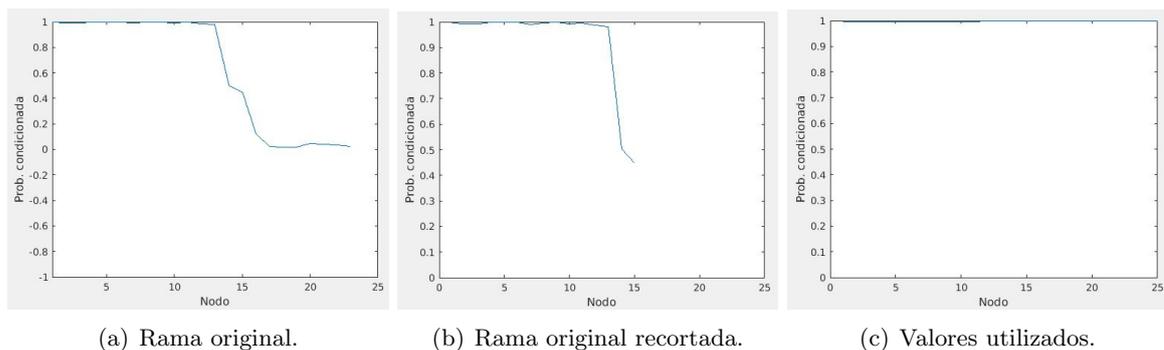


Figure 19: Representación de los nodos seleccionados.

En las imágenes puede verse la evolución de la rama. Como vemos se incluye el campo completo en el análisis (imagen h). En la siguiente tabla se puede ver como en función del área del nodo se asigna una probabilidad a posteriori de pertenecer a la clase patata. Para completar aquellos intervalo que no contienen ningún valor se intrapola empleando los valores conocidos.

Intervalo	1	11-30	31-80	81-115	116-345	346-633	634-917	918-1245	1246-1580
Área del nodo	1	22	75	-	129	430	-	-	1451
Prob. a posteriori	0.999	0.999	0.990	-	0.992	0.996	-	-	0.987
Prob. a posteriori estimada	0.999	0.999	0.990	0.991	0.992	0.996	0.992	0.987	0.987

Table 4: Probabilidad condicionada en función del área, tabla 1

Intervalo	1581-1841	1842-2164	2165-2470	2471-2651	2652-2877	2878-3163	3164-3450	3451-3742
Área del nodo	-	-	-	-	2799	-	-	-
Prob. a posteriori	-	-	-	-	0.981	-	-	-
Prob. a posteriori estimada	0.984	0.984	0.983	0.982	0.981	-1	-1	-1

Table 5: Probabilidad condicionada en función del área, tabla 2

Finalmente, la gráfica resultante es una aproximación de la gráfica inicial en la que la evolución de la probabilidad a posteriori se puede relacionar con el valor del área. Esto permite al clasificador comprender y adaptarse a ramas de diferentes tamaños.

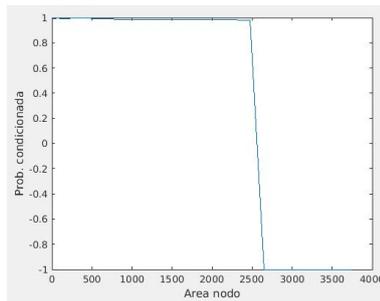


Figure 20: Resultado final.

### Análisis del modelo

Una vez creado el modelo con el árbol de decisión se puede calcular la importancia de los descriptores.

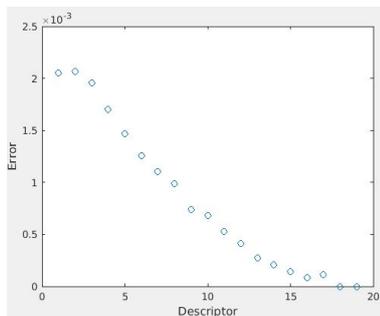


Figure 21: Importancia de los intervalos.

Observando la imagen se puede ver que los valores más importantes son los que comprenden áreas más pequeñas. Confirma que la idea de añadir divisiones extra para tener más información de los nodos cercanos a la hoja mejora el sistema. Se puede ver que la utilidad de los descriptores se va reduciendo a medida que se incrementa el área observada. Los dos últimos intervalos, que comprenden la región 3164-3742, no aportan información útil.

## Capítulo 4

### 4 Resultados

En este apartado se muestran los resultados obtenidos empleando los métodos obtenidos anteriormente. Así como se habla de la elección de parámetros obtenida a raíz de los resultados de las pruebas.

#### 4.1 Clasificación a nivel de píxel

Para la clasificación se emplean diferentes descriptores que definirán los píxeles. Una vez definidos los descriptores se aplicara un algoritmo de clasificación supervisado para crear el modelo de clasificación. Se realizaran las pruebas con SVM y con Árbol de clasificación

##### 4.1.1 Selección de descriptores

Los descriptores tenidos en cuenta han sido: Descomposición de Hoekman, matriz de coherencia  $\underline{T}$ , autovalores, Span, Entropía, Anisotropía y ángulo alpha.

Estos son los resultados obtenidos al emplear para la clasificación los descriptores por separado.

	<b>D. Hoekman</b>	<b>de coherencia <math>\underline{T}</math></b>	<b>Autovalores</b>	<b>H, A y alpha</b>
<b>Árbol</b>	0.795	0.794	0.759	0.646
<b>SVM</b>	0.809	0.815	0.775	0.570

Table 6: Fscore del descriptor.

Estos son los resultados empleando todos los descriptores exceptuando el indicado en la tabla.

	<b>D. Hoekman</b>	<b>Matriz de coherencia <math>\underline{T}</math></b>	<b>Autovalores</b>	<b>H, A y alpha</b>
<b>Árbol</b>	0.815	0.810	0.815	0.814
<b>SVM</b>	0.830	0.823	0.830	0.829

Table 7: Fscore al excluir el descriptor.

Como se puede apreciar en la tabla6, al eliminar la matriz de coherencia  $\underline{T}$  se reduce notablemente el indice de acierto del sistema para ambos clasificadores. Al eliminar los valores de Entropía, Anisotropía y angulo alpha el acierto también se reduce, pero en menor medida. Podemos deducir que estos descriptores aportan información que no encontramos en ninguna otra parte y que deben ser incluidos en la creación del modelo.

Por otro lado, al eliminar la descomposición de Hoekman o los autovalores, el indice de acierto no se ve afectado. Por lo tanto estos descriptores aportan información redundante.

El Fscore obtenido con un modelo empleando la matriz de coherencia  $\underline{T}$  y Entropía, Anisotropía y ángulo  $\alpha$  es el siguiente:

	<b>Matriz de coherencia <math>\underline{T}</math>, H, A, alpha</b>	<b>Todos los descriptores</b>
<b>Arbol</b>	0.810	0.815
<b>SVM</b>	0.827	0.830

Table 8: Fscore del modelo con matriz de coherencia T y Entropía, Anisotropía y ángulo  $\alpha$ .

El Fscore obtenido con estos dos descriptores es inferior al obtenido empleándolos todos. Para obtener el mejor resultado posible, aun es necesario emplear más descriptores.

Se crean dos modelos más, en uno se emplean los valores no descartados de la descomposición de Hoekman y se descartan los autovalores y en el otro se hace lo contrario.

	<b>Hoekman, <math>\underline{T}</math>, H, A y alpha</b>	<b><math>\underline{T}</math>, autovalores, H, A y alpha</b>
<b>Arbol</b>	0.815	0.815
<b>SVM</b>	0.830	0.830

Table 9: Comparación de los modelos.

Vemos que los resultados son iguales. Si descartamos algún descriptor más el índice de acierto se reduce. Podemos concluir que los valores no descartados de Hoekman y los autovalores representan la misma información aunque no sean redundantes componente a componente en conjunto si lo son. Las dos opciones son igualmente válidas y eficaces para la clasificación. Es indiferente cual utilicemos.

Se comprueba el resultado obtenido comparando el resultado de clasificación de los datos de test con el obtenido al crear el modelo para comprobar el sobreentrenamiento.

	<b>Conjunto de entrenamiento</b>	<b>Conjunto de test</b>
<b>Arbol</b>	0.810	0.815
<b>SVM</b>	0.850	0.830

Table 10: Fscore de la clasificación de los datos del modelo y de los datos de test.

Los resultados entre los datos empleados para crear el modelo y los de test son similares. El sistema no está sobreentrenado.

#### 4.1.2 Promediado de la imagen

Como se ha dicho en el capítulo 2.2, es necesario promediar los píxeles de la imagen. El tamaño del promediador influye en la clasificación, pero un promediador demasiado grande hace que perdamos resolución. Se ha de seleccionar uno que combine el mayor Fscore con el menor tamaño posible. Para encontrar el tamaño adecuado de promediador se aplicaron diversos filtros y se entrenaron varios modelos de clasificación (utilizando un Arbol de decisión), para luego medir el Fscore.

Tamaño filtro	3x3	5x5	7x7	9x9	11x11	13x13
Fscore	0.450	0.558	0.622	0.768	0.818	0.815

Table 11: Fscore por cada filtro.

Las pruebas concluyen que el tamaño adecuado es 11x11.

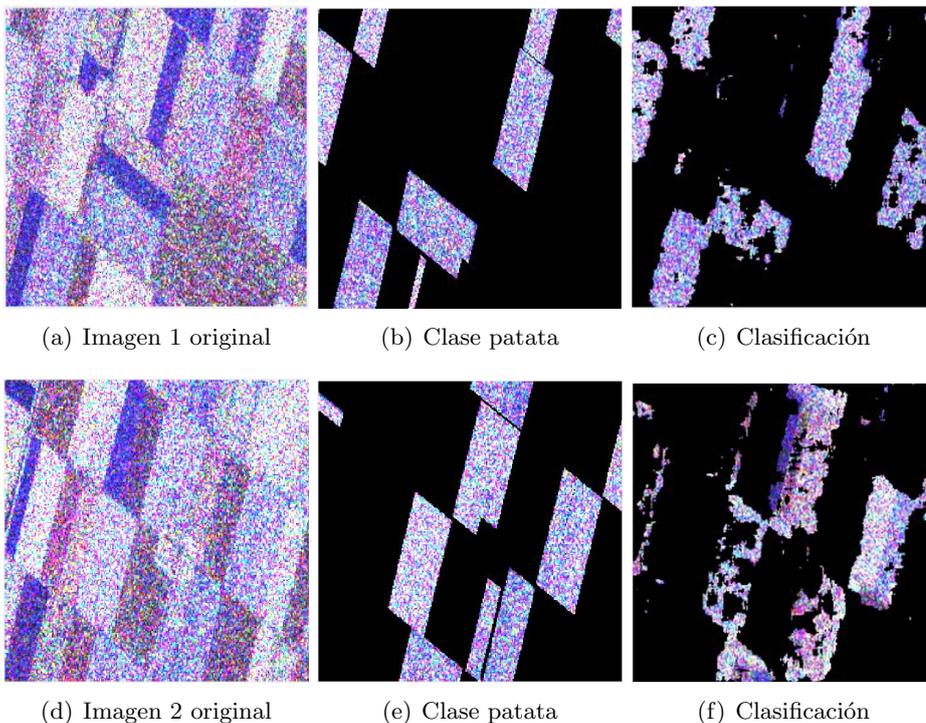
### 4.1.3 Resultados de la clasificación de pixeles

Para realizar el test de funcionamiento del sistema se emplean las imágenes mostradas en el apartado 3.2. El método de entrenamiento utilizado es SVM con kernel Gaussiano.

	Imagen 1	Imagen 2	Imagen 3	Imagen 4
Fscore	0.818	0.793	0.775	0.837

Table 12: Fscore de las imagenes de test.

Visualizando de los resultados de la clasificación se puede observar que hay muy pocos pixeles que sean clasificados como patata erróneamente. Es más común ver regiones dentro de los propios campos de patata que no son identificados como tales. Como dato a destacar, se puede apreciar que en la figura 1 se predice un campo entero como perteneciente en la clase patata, aun así no esta etiquetado como tal. Este campo no consta de ninguna etiqueta que nos diga su tipo.



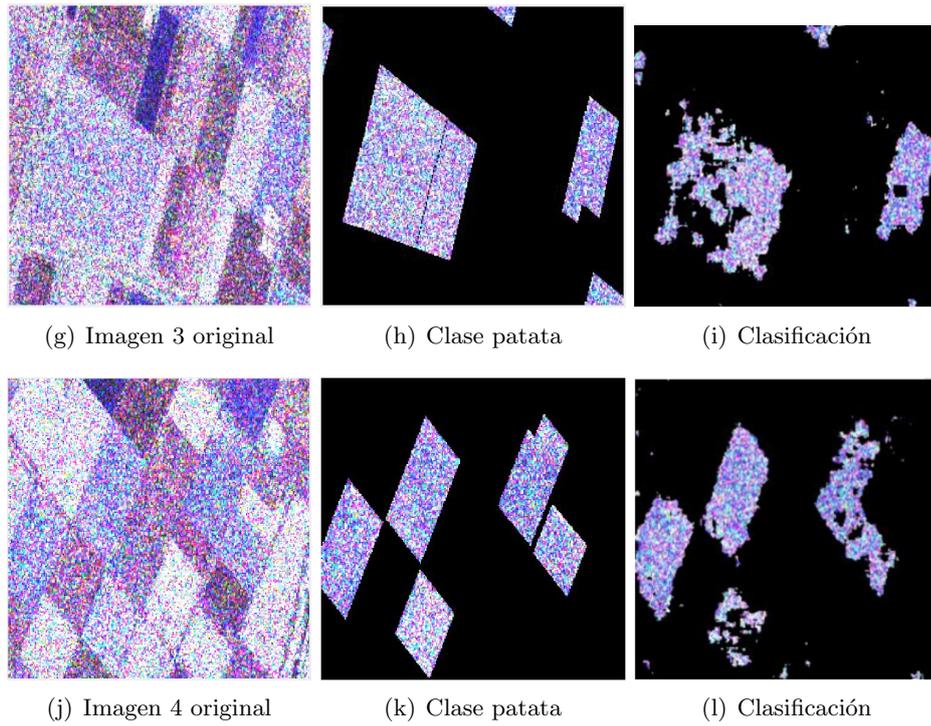


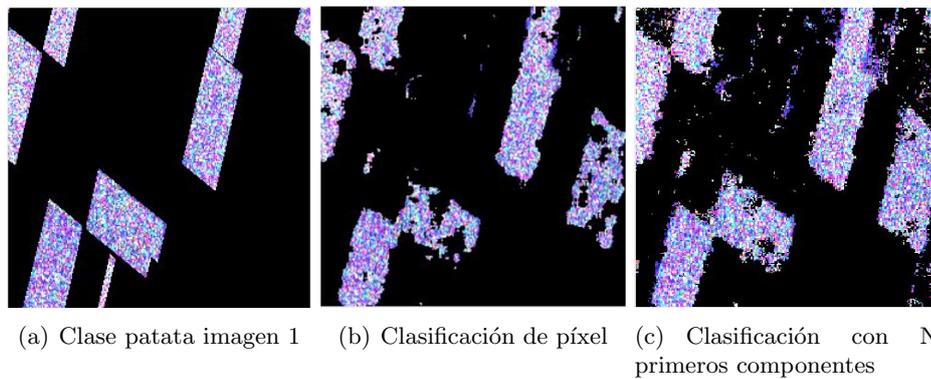
Figure 22: Clasificación de píxel.

## 4.2 Clasificación mediante BPT

En este apartado se muestran los resultados de la clasificación empleando el BPT.

### 4.2.1 N primeros componentes

Se visualizan los resultados de la clasificación para  $N=15$ .



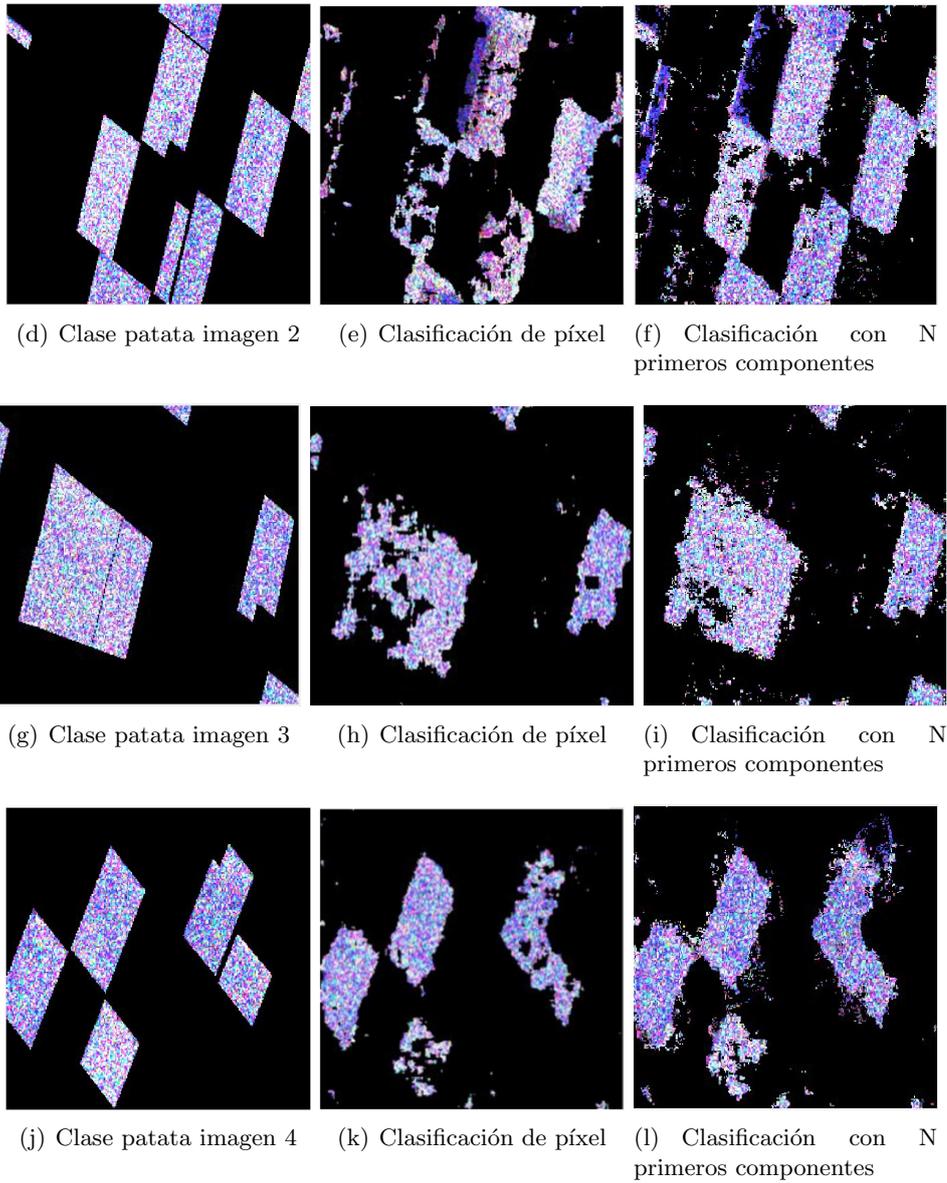


Figure 23: Comparación de la clasificación con N primeros componentes.

	Arbol N=10	Arbol N=15	Arbol N=20	Arbol N=25	Clasificación de píxel
<b>Imagen 1</b>	0.842	0.834	0.641	0.623	0.818
<b>Imagen 2</b>	0.809	0.846	0.719	0.675	0.793
<b>Imagen 3</b>	0.776	0.871	0.800	0.751	0.775
<b>Imagen 4</b>	0.812	0.865	0.708	0.673	0.837

Table 13: Fscore de la clasificación con N primeros componentes comparado con el resultado de la clasificación de píxel.

Obtenemos los mejores resultados con N=15.

#### 4.2.2 Probabilidad condicionada en función del área

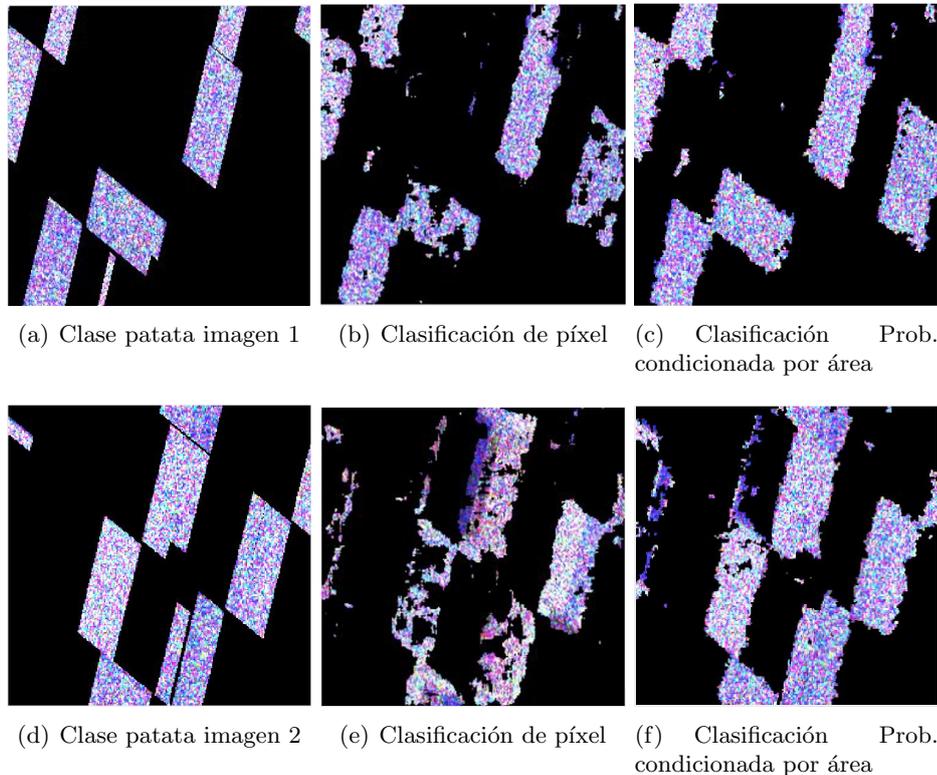
Al emplear el método de probabilidad condicionada en función del área obtenemos los siguientes resultados:

	Arbol	SVM	Clasificación de pixel
<b>Imagen 1</b>	0.891	0.856	0.818
<b>Imagen 2</b>	0.896	0.877	0.793
<b>Imagen 3</b>	0.923	0.912	0.775
<b>Imagen 4</b>	0.879	0.779	0.837

Table 14: Clasificación utilizando probabilidad condicionada en función del área comparado con el resultado de la clasificación de píxel.

En este caso el árbol de decisión presenta unos resultados mejores que SVM. Por lo tanto se considera que es un mejor sistema para este tipo de problema.

Se visualizan los resultados de la clasificación.



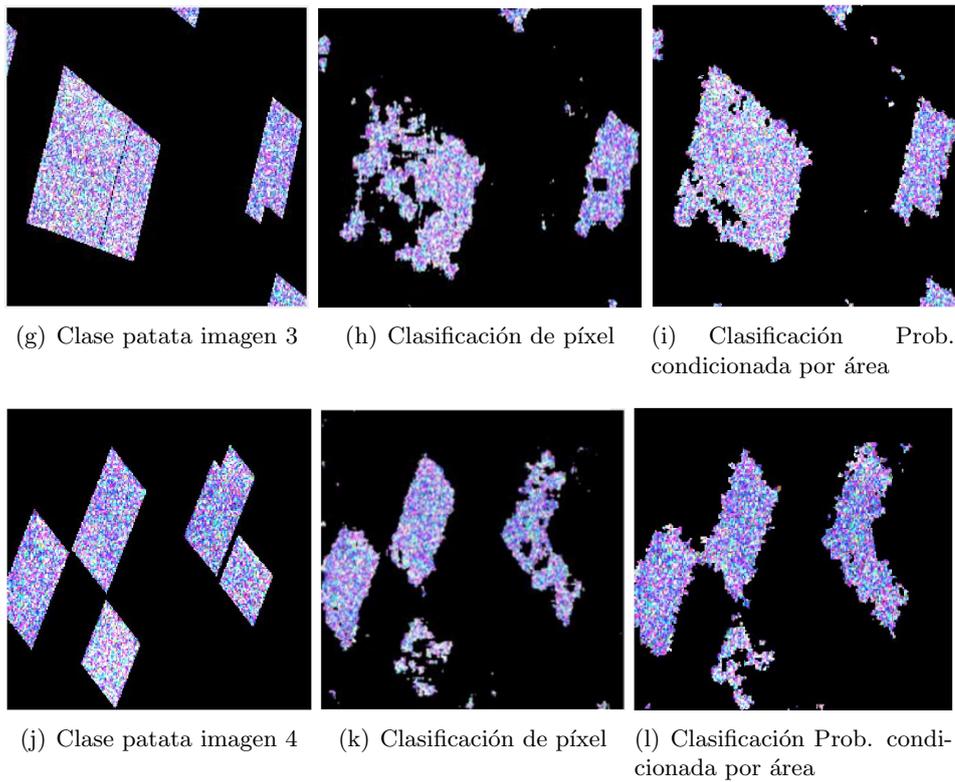


Figure 24: Comparación de las clasificaciones sobre las imágenes de test.

### 4.2.3 Análisis de los resultados

Comparando las 3 predicciones, la realizada empleando los píxeles, la realizada con los 15 primeros componentes de la rama y la probabilidad condicionada en función del área podemos ver en que situaciones el BPT puede ser útil para mejorar la clasificación.

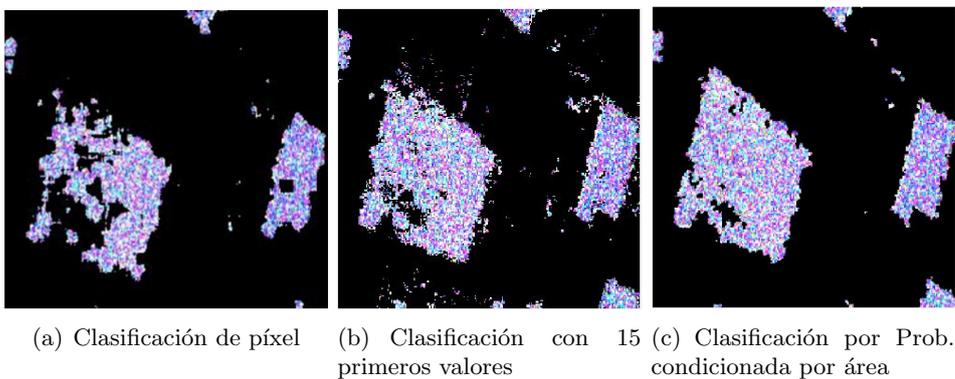


Figure 25: Clasificación de la imagen 3.

El primer caso a analizar es un píxel de patata que en los 3 casos ha sido representado de forma correcta.

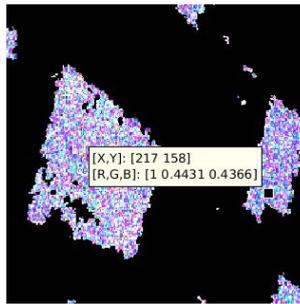


Figure 26: Píxel seleccionado

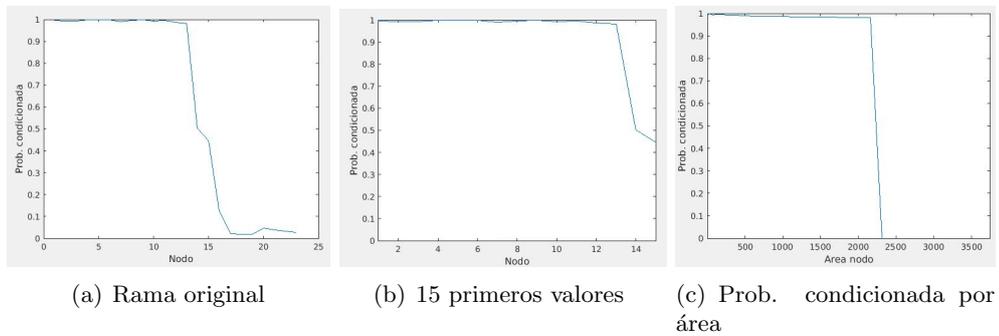


Figure 27: Representación de la rama

Todos los nodos cercanos a la hoja son clasificados como pertenecientes a la clase. El resto de píxeles obtenidos en regiones similares presentan la misma forma.

En este caso vemos un píxel que originalmente no había sido identificado como patata pero empleando el BPT es identificado correctamente.

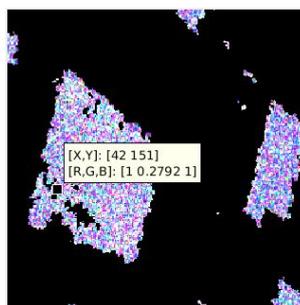


Figure 28: Píxel seleccionado

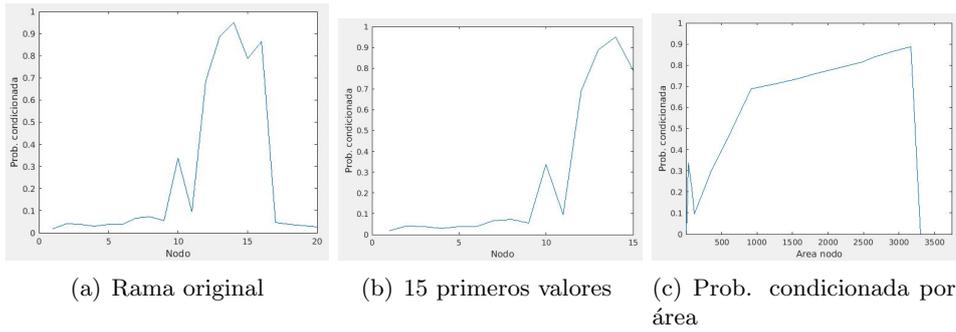


Figure 29: Representación de la rama

En este caso las regiones cercanas a la hoja no han sido identificadas correctamente. La identificación ha sido posible gracias a valores superiores de la rama.

En este caso vemos un píxel que en ningún momento es identificado correctamente.

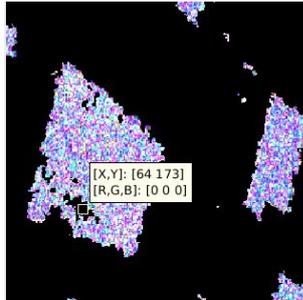


Figure 30: Píxel seleccionado

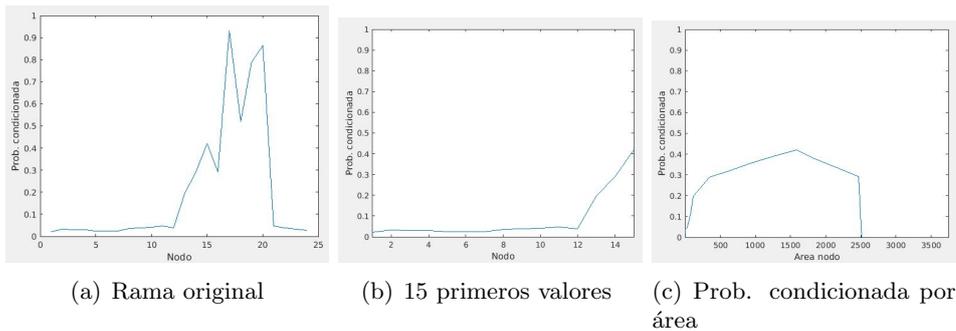


Figure 31: Representación de la rama

Sólo unos pocos nodos en toda la rama son identificados como pertenecientes a la clase. Emplear únicamente 15 componentes es insuficiente en este caso ya que se pierde información esencial. La probabilidad condicionada por área tampoco es suficiente ya que los intervalos superiores de la rama no están representados. Podemos considerar este caso como el peor posible, una región muy amplia no se identifica correctamente y al unirse al resto del campo al que pertenece el área es anormalmente grande, por lo tanto la información no está correctamente representada. Para realizar las pruebas se ha utilizado un dataset compuesto por una imagen

*RADARSAT*<sub>2</sub> Fine Quad-Pol imagenes perteneciente a

En este caso vemos un pixel sólo es identificado mediante el sistema de la probabilidad en función del área.

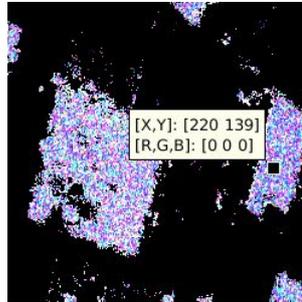


Figure 32: Píxel seleccionado

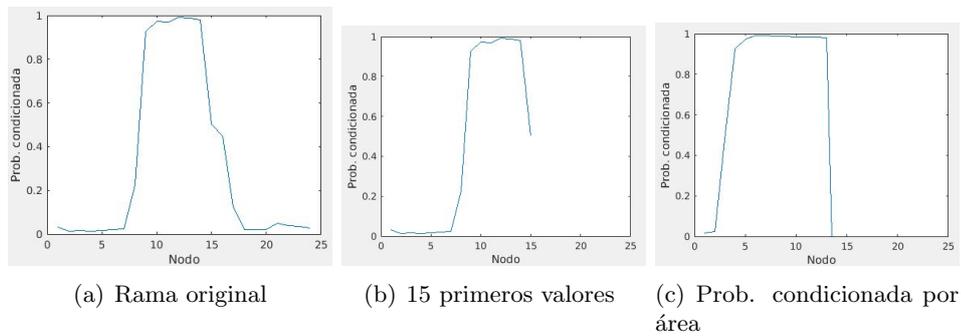


Figure 33: Representación de la rama

A pesar de que en  $N$  primeros componentes esta casi toda la información de la rama el modelo no es capaz de interpretarlo correctamente, esto demuestra que no es capaz de adaptarse a todos los tipos de rama. El modelo de probabilidad en función del área realiza correctamente la clasificación.

En este caso vemos un píxel que no pertenece a la clase patata, pero es identificado como tal en el método de  $N$  componentes.

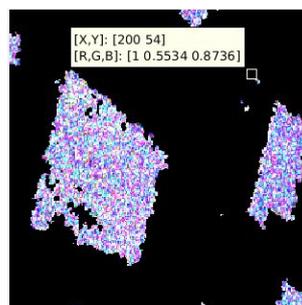


Figure 34: Píxel seleccionado

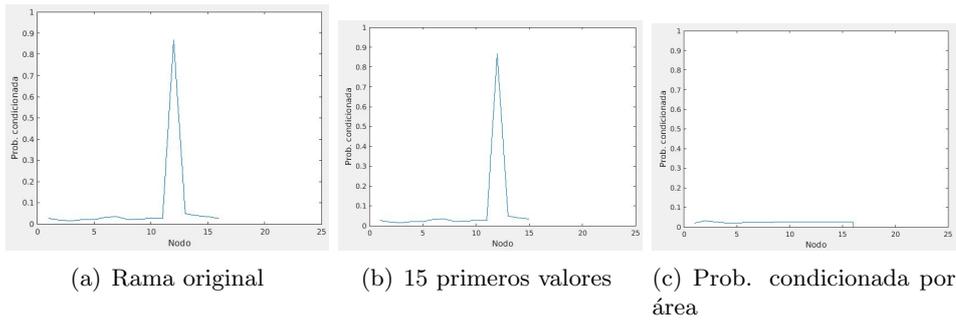


Figure 35: Representación de la rama

Casi toda la información de la rama esta comprendida en N primeros componentes. Igual que en el caso anterior, el modelo no es capaz de interpretar la información de la rama. El modelo de probabilidad condicionada en función de área no representa en ningún momento la posibilidad de que el pixel pertenezca a la clase patata.

## Capítulo 5

### 5 Conclusiones y posible líneas de trabajo futuro

En este capítulo se extrae la conclusión de los resultados y se proponen líneas de trabajo futuro.

#### 5.1 Conclusiones

El objetivo de este trabajo era comprobar si se podía emplear BPT para mejorar la clasificación realizada a nivel de píxel. Los resultados demuestran que es una herramienta útil para ese propósito. El BPT proporciona información, definiendo la rama de cada píxel, que pueden ser utilizada para la identificación.

El modelo de clasificación a nivel de píxel puede emplearse para clasificar los nodos. Pero se obtienen mejores resultados si también se incluyen nodos en el entrenamiento. La probabilidad condicionada del nodo es un parámetro que nos permite describir con precisión el resultado de la clasificación y representar una rama en un único vector.

El análisis los modelos de clasificación propuestos que usan la información de la rama corrobora que los nodos más útiles son los más cercanos a la hoja. Dado que los nodos donde se produce la mezcla son los de la parte superior, es lógico suponer que podemos prescindir de la parte superior de la rama.

El sistema de  $N$  primeros componentes con  $N=15$  ignora toda información a partir del nodo 15. En casos donde la rama tiene menos de 15 componentes se rellena con valores que no aportan ninguna clase de información (-1). El problema de este sistema es que dependiendo del tamaño de la rama, se ignora información que puede ser crucial para la identificación y que el modelo no puede adaptarse a todos los tamaños de rama, por lo que incluso en casos donde aparentemente la información contenida en los 15 primeros valores debería ser suficiente puede no serlo. Aun así este sistema consigue mejorar la clasificación.

El sistema de probabilidad condicionada en función del área se basa en que la probabilidad condicionada cambia en función de la cantidad píxeles añadida de un nodo a otro. Se puede hacer una aproximación lineal del valor de la probabilidad condicionada entre dos nodos conocidos. Al definir las áreas de los intervalos, se puede escoger el tamaño de los nodos que queremos tener en cuenta. Lo cual nos permite ignorar valores de la rama que no nos aportarían información útil. Este método ha demostrado ser capaz de representar de forma uniforme la información sin importar el tamaño de la rama. Este es el mejor método para la clasificación.

#### 5.2 Posibles líneas de trabajo futuro

Para mejorar los resultados de clasificación se podría probar a utilizar imágenes de la misma región en diferentes meses del año. Esto podría aumentar el Fscore de la clasificación de nodos y por lo tanto incrementar el acierto tanto a nivel de píxel como empleando el BPT.

# Bibliográfica

## References

- [1] Alberto Alonso-Gonzalez. Multidimensional and temporal SAR data representation and processing based on Binary Partition Trees. *Thesis for the PhD in Signal Theory and Communications*, May 2014.
- [2] Alberto Alonso-Gonzalez, Carlos Lopez-Martnez, and Philippe Salembier. Filtering and Segmentation of Polarimetric SAR Data Based on Binary Partition Trees. *IEEE transactions on Geoscience and Remote Sensing*, 50(2):593-605, February 2012.
- [3] Cristian Cuadrado Conde. PolSAR image classification with Binary Partition Tree, Project Critical Review. *A Degree's thesis documentation*, November 2016.
- [4] European Spatial Agency. Polarimetry Tutorial, Polarimetric Decomposition <https://earth.esa.int/web/polsarpro/polarimetry-tutorial>
- [5] John C. Curlander and Robert N. McDonough. *Synthetic Aperture Radar: systems and signal processing*, volume XVII of *Wiley Series in Remote Sensing*. John Wiley & Sons, Inc., 1991.
- [6] Dirk H. Hoekman. A new polarimetric classification approach evaluated for agricultural crops. *IEEE Transactions on Geoscience and Remote Sensing* (Volume: 41, Issue: 12, Dec.3)
- [7] Nghiem, S.V., S.H. Yueh, R. Kwok, and F.K. Li, 1992, Symmetry properties in polarimetric remote sensing, *Radio Science*, Vol.27, pp.693-711.
- [8] Philippe Salembier and Luis Garrido. Binary Partition Tree as an efficient Representation for Image Processing, Segmentation, and Information Retrieval. *IEEE Transactions on Image Processing*, 9(4):561-576, April 2000.
- [9] Amanpreet Singh, Narina Thakur and Aakanksha Sharma. A Review of Supervised Machine Learning Algorithms. *Computing for Sustainable Global Development (INDIACom)*, 2016 3rd International Conference on 16-18 March.
- [10] Jean-Philippe Vert, Koji Tsuda, and Bernhard Schölkopf. "A primer on kernel methods". *Kernel Methods in Computational Biology*.
- [11] Yu Bai, Wen Yang<sup>1</sup>, Gui-Song Xia, Mingsheng Liao. A Novel Polarimetric-Texture-Structure Descriptor for High-Resolution PolSAR Image Classification, *IEEE* 26-31 July 2015
- [12] Joel Max, Quantizing for Minimum Distortion, *IRE Transactions on Information Theory*, Vol. IT-6, March, 1960, pp. 7–12.

# Lista de acrónimos

SAR	Synthetic Aperture Radar
PolSAR	Polarimetric Synthetic Aperture Radar
BPT	Binary Partition Tree
BSA	Back Scattering Alignment
S,k	Matriz/Vector de dispersión
C	Matriz de covarianca (Base Lexicografica)
T	Matriz de coherencia (Bases de Pauli)
SVM	Support Vector Machine
RBF	Radial Basis Function

# Colofón

Esta tesis se ha escrito utilizando ,  $\text{\LaTeX}$  desarrollado originalmente por Leslie Lamport y basado en  $\text{\TeX}$  de Donald Knuth. La plataforma utilizada ha sido  $\text{\TeX}$ Maker.