



Máster en Ingeniería de Sistemas y Control

**Identificación de estructuras vegetales en campos de cultivo de maíz**

Realizado por:

Manuel David Franco Barrios

Dirigido por:

Gonzalo Pajares Martinsanz

María Guijarro Mata-García

Curso académico 2011/2012

Convocatoria de Septiembre

Máster en Ingeniería de Sistemas y Control

Proyecto Fin de Máster

Septiembre 2012

**Identificación de estructuras vegetales en campos de cultivo de maíz**

Realizado por:

Manuel David Franco Barrios

Dirigido por:

Gonzalo Pajares Martinsanz

María Guijarro Mata-García

*(espacio reservado para calificaciones)*

## Autorización

Autorizamos a la Universidad Complutense y a la UNED a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o el prototipo desarrollado.

Firmado:

## Resumen

A nivel general, se trata de un proyecto en el marco de la **Visión por Computador**, enfocado en particular en el ámbito del **reconocimiento de texturas**. El tema concreto estudiado es el de la **segmentación automática de texturas en imágenes agrícolas**.

Existe actualmente una necesidad general de automatizar las tareas agrícolas, y en los procedimientos empleados para ello se utilizan robots y máquinas cuyos sensores ópticos tienen un papel muy importante. Las imágenes que ofrecen dichos sensores deben ser procesadas adecuadamente, y en concreto una de las necesidades más importantes, que es la problemática aquí tratada, es la de identificar las plantas dentro de dichas imágenes, separándolas de otros elementos no relevantes, como pueden ser cielo y tierra, entre otros.

El proyecto trata el problema de la identificación de estructuras vegetales en imágenes agrícolas, en concreto de campos de cultivo de maíz, para su posterior tratamiento en el contexto de los procesos de automatización necesarios en lo que se conoce como Agricultura de Precisión, destacan por ejemplo la aplicación de productos químicos, tales como herbicidas, o el empleo de elementos mecánicos o de otra índole como el quemado de las malas hierbas.

## Palabras clave

*Visión por computador, Reconocimiento de texturas vegetales, Segmentación de texturas vegetales, Imágenes agrícolas, Campos de cultivo, Agricultura de Precisión, Automatización, Espacios de color, CIELab, Filtros de Gabor, Clustering, K-Means*

*A Marta, a mis amigos, y a mi familia, por el tiempo robado.*

## Índice de contenido

1. Descripción del problema y objetivos .....	10
2. Estado del arte.....	11
3. Alternativa planteada.....	12
3.1. Espacios de color.....	12
3.1.1. Percepción humana del color.....	13
3.1.2. Espacios de color perceptualmente uniformes.....	14
3.2. Espacios de características.....	16
3.2.1. Filtros de Gabor.....	16
3.2.2. Implementación como matrices de convolución.....	20
3.3. Formación de vectores de características.....	21
3.4. Agrupamiento mediante K-Means.....	22
4. Experimentaciones realizadas y resultados observados.....	25
4.1. Ejemplo detallado de clasificación.....	27
4.2. Otros ejemplos.....	39
4.3. Comparación de resultados respecto al método de referencia .....	41
5. Conclusiones.....	45
6. Trabajo futuro.....	47
7. Listado de acrónimos.....	48
8. Referencias .....	49
Anexo A: Código fuente .....	50



## Índice de figuras

Figura 1: Componentes de una función de Gabor en 1-D.....	17
Figura 2: Componentes de una función de Gabor en 2-D.....	17
Figura 3: Ejemplos de filtros de Gabor.....	18
Figura 4: Filtro de energía de Gabor en 1-D.....	19
Figura 5: Convergencia de los centros en algoritmo K-Means.....	23
Figura 6: Idea básica inicial del método.....	25
Figura 7: Descripción gráfica del método propuesto.....	27
Figura 8: Imagen original para ejemplo de aplicación.....	27
Figura 9: Canal L en CIELab.....	28
Figura 10: Canal a en CIELab.....	28
Figura 11: Canal b en CIELab.....	28
Figura 12: Filtro Gabor $\varphi=0$ y $\theta=1$ .....	29
Figura 13: Filtro Gabor $\varphi=\pi/4$ y $\theta=1$ .....	29
Figura 14: Filtro Gabor $\varphi=\pi/2$ y $\theta=1$ .....	29
Figura 15: Filtro Gabor $\varphi=3\pi/4$ y $\theta=1$ .....	29
Figura 16: Filtro Gabor $\varphi=0$ y $\theta=1.5$ .....	29
Figura 17: Filtro Gabor $\varphi=\pi/4$ y $\theta=1.5$ .....	29
Figura 18: Filtro Gabor $\varphi=\pi/2$ y $\theta=1.5$ .....	29
Figura 19: Filtro Gabor $\varphi=3\pi/4$ y $\theta=1.5$ .....	29
Figura 20: Respuesta de un filtro de Gabor con $\lambda=8$ , $\theta=1.5$ , $\varphi=0$ , sobre canal b.....	30
Figura 21: Respuesta de un filtro de Gabor con $\lambda=16$ , $\theta=1.0$ , $\varphi=\pi/2$ , sobre canal a.....	30
Figura 22: Respuesta de un filtro de Gabor con $\lambda=8$ , $\theta=1.5$ , $\varphi=3\pi/4$ , sobre canal a.....	30
Figura 23: Respuesta de un filtro de Gabor con $\lambda=4$ , $\theta=1.0$ , $\varphi=\pi/4$ , sobre canal b.....	30
Figura 24: Agrupamiento en 3 clases.....	31
Figura 25: Superposición del agrupamiento en 3 clases a la imagen original.....	31
Figura 26: Agrupamiento en 4 clases.....	32
Figura 27: Superposición del agrupamiento en 4 clases a la imagen original.....	32
Figura 28: Agrupamiento en 6 clases.....	33
Figura 29: Superposición del agrupamiento en 6 clases a la imagen original.....	33
Figura 30: Agrupamiento en 8 clases.....	33
Figura 31: Superposición del agrupamiento en 8 clases a la imagen original.....	33
Figura 32: Píxeles clasificados como terreno sin vegetación.....	34
Figura 33: Píxeles clasificados como cultivo.....	35
Figura 34: Píxeles clasificados como pastos.....	35
Figura 35: Píxeles clasificados como hierbas.....	36
Figura 36: Píxeles clasificados como arboleda.....	36
Figura 37: Píxeles sin interés, tipo I.....	37
Figura 38: Píxeles sin interés, tipo II.....	37
Figura 39: Píxeles sin interés, tipo III.....	38
Figura 40: Imagen con poca iluminación.....	39
Figura 41: Clasificación resultante.....	39
Figura 42: Surcos cultivados.....	39
Figura 43: Otra vegetación.....	39

Figura 44: Imagen con iluminación no uniforme.....	40
Figura 45: Clasificación resultante.....	40
Figura 46: Surcos cultivados.....	40
Figura 47: Otra vegetación.....	40
Figura 48: Imagen original de referencia.....	41
Figura 49: Clasificación de referencia.....	41
Figura 50: Clasificación en ocho grupos.....	42
Figura 51: Vegetación tipo I.....	42
Figura 52: Vegetación tipo II.....	42
Figura 53: Vegetación tipo III.....	42
Figura 54: Vegetación tipo IV.....	43
Figura 55: Vegetación tipo V.....	43
Figura 56: Unión de vegetación tipos I, II y III.....	43

## 1. Descripción del problema y objetivos

El propio título del proyecto, "*Identificación de estructuras vegetales en campos de cultivo de maíz basado en imágenes*", es muy descriptivo y resume la idea general del mismo. A nivel general, se trata de un proyecto en el marco de la **Visión por Computador**, enfocado en particular en el ámbito del **reconocimiento de texturas**. El tema concreto estudiado es el de la **segmentación automática de texturas en imágenes agrícolas**, cuyos precedentes se encuentran descritos en [1].

Tal y como se explica en dicho artículo, que reproduzco aquí para mayor comodidad del lector, existe actualmente una necesidad general de automatizar las tareas agrícolas, y en los procedimientos empleados para ello los sensores ópticos tienen un papel muy importante. Las imágenes que ofrecen dichos sensores deben ser procesadas adecuadamente, y en concreto una de las necesidades más importantes, que es la problemática aquí tratada, es la de identificar las plantas dentro de dichas imágenes, separándolas de otros elementos no relevantes, como pueden ser cielo y tierra, entre otros.

En el caso concreto de nuestro problema, que es el planteado en dicho artículo, se trata de imágenes tomadas sobre campos de maíz, y se necesita incluso separar los distintos tipos de plantas, como serían el maíz y las malas hierbas, con el objetivo de llevar a cabo algún tipo de tratamiento físico sobre las segundas, como la aplicación de herbicidas o la quema de las mismas mediante un brazo robótico. Este último es de hecho un caso real de aplicación dentro del proyecto de investigación referenciado en [2].

Expuesta la problemática, el objetivo establecido para el presente proyecto era ~~de tratar~~ proponer una solución alternativa a la planteada en el propio artículo [1] con el fin de tratar ~~de tratar~~ de identificar alguna problemática no descubierta, a la vez que mejorar en lo posible los resultados. Como material, además del propio artículo en sí, se dispone de un conjunto de imágenes de muestra sobre las que experimentar, tomadas como ya se ha dicho sobre campos de maíz, de forma que en todas ellas aparecen vegetales, tierra, cielo y algunas otras estructuras como edificios varios.

En resumen, el objetivo del proyecto era proponer un método para, dentro de las imágenes provistas, ser capaz de:

1. **Identificar la vegetación**, aislándola del resto de elementos no relevantes.
2. Dentro de las estructuras vegetales identificadas, **discriminar la planta de las malas hierbas**.

## 2. Estado del arte

En el trabajo [1] se exponen, como ya se ha indicado, tanto el problema como una solución al mismo que consiste, a grandes rasgos, en calcular una serie de índices para cada píxel a partir de la información proporcionada por su color, componiendo una nueva imagen por cada índice, y luego fusionando la información de las mismas aplicando varios métodos de umbralización o "thresholding". Dichos índices: ExG, ExR, CIVE, ExGR; ya se mencionaban en otros artículos referenciados por [1] y anteriores al mismo, pero se propone una forma de combinarlos para obtener mejor fiabilidad a la hora de separar plantas, tierra y cielo.

En [1] no se contempla el uso de análisis de frecuencia alegando que las características espectrales de los distintos tipos de vegetación a discriminar son muy similares, y que esto, unido a que las imágenes son tomadas al aire libre y expuestas por tanto a condiciones muy adversas como la exposición a diferentes condiciones de iluminación, sombras, o condiciones climáticas variables, factores todos que pueden afectar la composición espectral, dificulta mucho el empleo de métodos basados en esta técnica. En [3] y [4] se habla de métodos basados en el análisis de píxeles individuales y métodos basados en regiones, descartando los segundos por asumir que en las imágenes aéreas sobre las que se centran ambos trabajos no existen patrones identificables como texturas.

No obstante sí que existen en la literatura estudios sobre métodos basados en el análisis espectral para la clasificación y segmentación de texturas [5], incluso en imágenes naturales [6] [7]. De hecho es ampliamente utilizado el análisis mediante filtros de Gabor [5] [6] [7] [8] [9] [10] [11] [12] [13].

La referencia más importante a este respecto es [6], que trata concretamente del análisis de texturas en imágenes multispectrales, y en particular de la clasificación de texturas no homogéneas en imágenes naturales, tratándose por tanto de un problema de la misma naturaleza que el nuestro. El método que proponen hace hincapié en la correspondencia con la percepción visual humana a la hora de realizar la clasificación, aunque curiosamente no utilizan un espacio de color perceptualmente uniforme, sino que se basan en la aplicación de filtros de Gabor multi-escala sobre las componentes de la imagen en el espacio de color HSI.

También para el análisis de imágenes aéreas en [14] se asume la variabilidad o ambigüedad de las características extraíbles para la identificación de texturas, y proponen la aplicación de la teoría de conjuntos difusos para afrontar la misma.

Por último, en [10] se proponen una serie de medidas de similitud o "distancia" entre texturas caracterizadas mediante filtros de Gabor, que podrían ser de utilidad comparar a la hora de clasificar las imágenes.



### 3. Alternativa planteada

De entrada, y una vez considerados ~~todos~~ los métodos mencionados en la sección anterior, se optó por explorar un camino lo más diferente posible de la solución expuesta en el artículo de referencia [1], de forma que la comparación de los resultados fuera más interesante.

Así, el primer punto de distanciamiento fue basar la clasificación de cada píxel de la imagen en un conjunto de características que describan el entorno del mismo, en contraste con el método empleado en [1] que utiliza exclusivamente la información de color de cada píxel individualmente para calcular los índices que luego utiliza para segmentar, si bien es verdad que este proceso de segmentación ya incluye información global de la imagen, pero en ningún momento se examina el entorno local de cada píxel. De hecho, tanto en [1] como en [3] y [4] se desaconseja esta aproximación por la gran variabilidad existente en las texturas presentes en imágenes naturales y aéreas según el caso.

No obstante, un reconocimiento de texturas propiamente dicho podría plantearse mediante el estudio de regiones y no píxeles aislados, ya que de acuerdo al concepto de textura [15], ~~esta~~ no existe en un único punto, sino que una textura se caracteriza por la distribución de los valores de un conjunto de píxeles vecinos. Teniendo además en cuenta que se quieren distinguir diferentes tipos de estructuras vegetales, puede resultar difícil diferenciar las mismas basándonos exclusivamente en la tonalidad de color, pues siempre van a existir vegetaciones diferentes con colores casi idénticos al estar jugando con una gama muy estrecha del espectro de color.

Por lo expuesto, una alternativa para tratar de mejorar la identificación propuesta consiste en intentar controlar la problemática variabilidad presente en las texturas vegetales, sin dejar de estudiarlas como tales, es decir, intentando identificar texturas en regiones locales.

Para ello, se ha empleado una combinación de técnicas ya descritas en las referencias mencionadas en la sección anterior, siendo las más importantes una selección apropiada del espacio de color sobre el que trabajar, descripción de texturas mediante filtros de Gabor, y agrupamiento de las mismas en diferentes clases. En las siguientes subsecciones se exponen los conceptos fundamentales de cada uno de estos elementos por separado, y en la sección posterior se describe su uso conjunto en el método propuesto.

#### 3.1. Espacios de color

Según [6] la elección del espacio de color es esencial en el análisis de texturas en imágenes multiespectrales.

En esta sección se presentan los conceptos básicos para entender qué es un espacio de color, cuáles

son los más ampliamente usados, cuáles son sus características, y cuál ha sido la elección para el presente proyecto.

### *3.1.1 Percepción humana del color*

La percepción humana del color se basa en la incidencia de la luz visible (ondas electromagnéticas cuya longitud de onda varía de 400nm a 700 nm) en la retina, en la cual existen tres tipos de células foto-receptoras, cada uno con una curva de respuesta espectral diferente, por lo que en teoría todos los colores pueden especificarse mediante tres números que corresponderían a las salidas de dichos fotorreceptores.

En 1931 la "Commission Internationale de l'Eclairage" (CIE) estandarizó las curvas de respuesta de los fotorreceptores de un hipotético observador estándar, definió el sistema CIE XYZ mediante el que cualquier color visible puede ser representado mediante valores positivos de X, Y, y Z. Puede verse, por tanto, dicho sistema XYZ como un espacio en tres dimensiones donde cada vector define un color, teniendo aquí en consecuencia la definición de espacio de color.

Dicho espacio XYZ comprende como se ha dicho todos los colores visibles, sin embargo no es el único espacio de color estándar existente, sino que son muy utilizados una serie de subespacios del mismo diseñados con propósitos específicos. Los más comunes son:

- RGB: es un sistema de colores aditivo basado en la suma de los tres colores primarios rojo, verde y azul, utilizado comúnmente en pantallas CRT.
- CMYK: sistema de color sustractivo usado en imprenta, donde la ausencia de color, origen del espacio, es el blanco.
- HSL: sistema de colores diseñado para ser intuitivo a la hora de permitir a un usuario especificar un color fácilmente. La componente de luminosidad está separada en un canal propio, lo que tiene ventajas para ciertas aplicaciones de procesamiento de imágenes.

Todos ellos son subespacios diferentes del espacio XYZ, lo que quiere decir que cada uno de ellos deja fuera un conjunto de colores que no puede representar.

Además, al estar compuesta la base de cada espacio por un conjunto diferente de vectores en XYZ, la distancia existente entre dos colores dados es diferente según se mida en uno u otro espacio. Esto es muy relevante de cara a muchas aplicaciones de tratamiento de imágenes, entre ellas la nuestra.

### 3.1.2 Espacios de color perceptualmente uniformes

En ciertas aplicaciones es muy importante poder medir la distancia entre colores de forma que se corresponda lo más fielmente posible con la similitud entre esos colores percibida por un ser humano, para lo cual han sido diseñados una serie de espacios de color llamados "perceptualmente uniformes" o simplemente "uniformes".

Un espacio de color es perceptualmente uniforme si un pequeño desplazamiento de un color C en dicho espacio provoca un mismo grado de variación en la percepción independientemente de en qué lugar del espacio esté dicho color. Por ejemplo, en un espacio no uniforme como RGB, la diferencia perceptible entre un par de colores A y B (p.e. violeta y rojo) puede ser equivalente a una mayor diferencia perceptible entre B y C (rojo y negro), existiendo la misma distancia entre los respectivos pares ( $AB = BC$ ).

En 1976, la CIE estandarizó dos espacios de este tipo, llamados *Luv* (o CIELuv) y *Lab* (o CIELab), compuestos ambos por tres canales en los que el canal "L" contiene la información relativa a la luminosidad, y los otros dos ("u" y "v", o "a" y "b", respectivamente) con la relativa a la cromaticidad. Los dos espacios son también subespacios de CIE XYZ, y comparten un blanco de referencia  $X_n Y_n Z_n$ .

Dado que la gran mayoría de librerías existentes para el tratamiento de imágenes carga las mismas por defecto en espacio RGB, para transformarlas a espacio CIELab se pasa primero por el espacio de referencia XYZ, mediante las siguientes transformaciones:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.240479 & -1.537150 & -0.498535 \\ -0.969256 & 1.875992 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (2)$$

Para la transformación entre XYZ y CIELab o CIELuv, la luminosidad se define similarmente como sigue:

$$L = \begin{cases} 116(Y/Y_n)^{1/3} - 16 & \text{si } Y/Y_n > 0.008856 \\ 903.3(Y/Y_n) & \text{en otro caso} \end{cases} \quad (3)$$

Los canales de cromaticidad sí que se definen de forma diferente en CIELuv y CEILab. En el primer caso:



$$u = 13L(u' - u'_n) \quad (4)$$

$$v = 13L(v' - v'_n) \quad (5)$$

$$\begin{aligned} u' &= \frac{4X}{X + 15Y + 3Z} \\ v' &= \frac{9Y}{X + 15Y + 3Z} \end{aligned} \quad (6)$$

donde  $u'_n$  y  $v'_n$  se definen igual que  $u'$  y  $v'$  pero usando  $X_n$ ,  $Y_n$  y  $Z_n$ .

En el caso de CIELab:

$$a = 500(f(X/X_n) - f(Y/Y_n)) \quad (7)$$

$$b = 200(f(Y/Y_n) - f(Z/Z_n)) \quad (8)$$

$$f(t) = \begin{cases} t^{1/3} & \text{si } Y/Y_n > 0.008856 \\ 7.787t + 16/116 & \text{en otro caso} \end{cases} \quad (9)$$



### 3.2. Espacios de características

Como ya se ha expuesto previamente, el color es una propiedad a nivel de píxel, si bien la textura implica en sí un espacio o región, y no existe textura en un solo píxel [7] [15].

Para describir la textura en una región es normal utilizar una proyección de la función de intensidad de la imagen sobre un conjunto de funciones. Esto se suele denominar descomposición espectral, dado que cada una de dichas funciones suele medir la intensidad en un área diferente del dominio de la frecuencia espacial en dos dimensiones, aunque puede haber otras alternativas.

La escala, la frecuencia espacial, y la orientación son características muy importantes de las texturas presentes en una imagen, y son las que queremos extraer o identificar en el marco del trabajo que se propone. En nuestro caso, una textura se describe como un vector de valores, donde cada uno corresponde a la energía relativa en una subbanda de frecuencia, de escala y de orientación específica.

El método de descomposición espectral que empleamos está basado en los denominados filtros de Gabor [5] [6] [7] [8] [9] [11] [12] [13] [16]. Existe literatura que evidencia cómo puede modelarse mediante filtros de Gabor la respuesta de las neuronas de la corteza visual primaria del cerebro de algunos animales. Incluso en [16] proporcionan algunas guías sobre cómo modelarlas, aunque nosotros utilizaremos un banco mucho más simple, que veremos será suficiente como prueba de concepto.

#### 3.2.1. Filtros de Gabor

Una función de Gabor es una función senoide modulada por una gaussiana. En las figuras 1 y 2 se muestra de forma intuitiva el concepto en el caso de una y de dos dimensiones.

En el caso unidimensional, figura 1, una función como esta maximizará su respuesta en una frecuencia específica, dependiendo de la componente sinusoidal, y sólo en una parte localizada de la señal, dependiendo de la componente gaussiana. En el caso bidimensional, figura 2, la salida dependerá también de la orientación de la senoide.

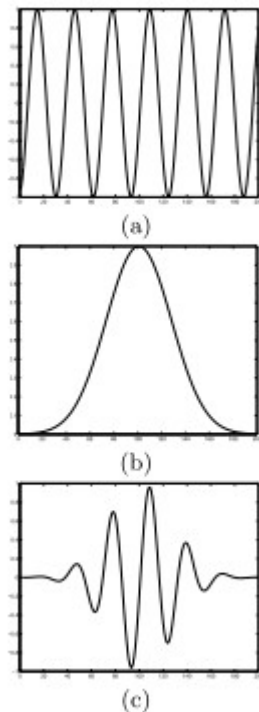


Figura 1: Componentes de una función de Gabor en 1-D

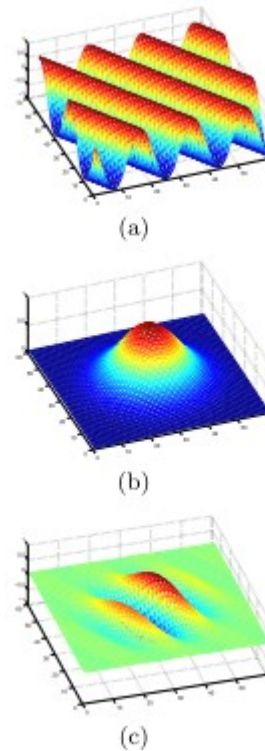


Figura 2: Componentes de una función de Gabor en 2-D

Sea  $g(x, y, \lambda, \theta, \phi)$  una función de Gabor con un escala  $\lambda$ , frecuencia  $\theta$  y orientación  $\phi$ , definida mediante la siguiente fórmula:

$$g(x, y, \lambda, \theta, \phi) = e^{-\frac{x^2 + y^2}{(\lambda/2)^2}} e^{j 2\pi \frac{\theta}{\lambda} (x \cos \phi + y \sin \phi)} \quad (10)$$

Nótese que la primera exponencial corresponde a la componente gaussiana, y la segunda define la senoide.

La figura 3 muestra la representación en tres dimensiones de algunas funciones de Gabor, y su aspecto al representarlas en el plano de una imagen, parte derecha.

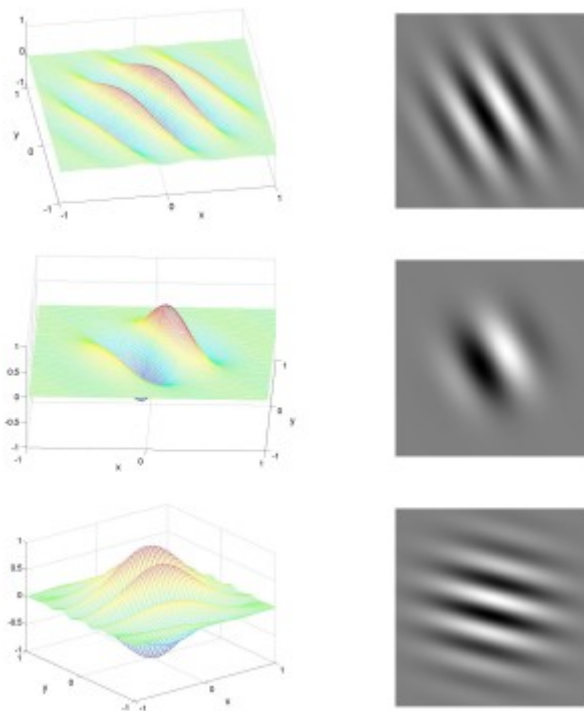


Figura 3: Ejemplos de filtros de Gabor



La respuesta de una función de Gabor a una imagen se obtiene como una función de convolución bidimensional. Sea  $I(x,y)$  la función de intensidad de la imagen y  $G(x, y, \lambda, \theta, \phi)$  la respuesta del filtro de Gabor con escala  $\lambda$ , frecuencia espacial  $\theta$  y orientación  $\phi$  a una imagen en el punto  $(x, y)$ ,  $G$  se obtiene como sigue:

$$G(x, y, \lambda, \theta, \phi) = \iint I(p, q) g(p-x, q-y, \lambda, \theta, \phi) dp dq \quad (11)$$

En realidad, tal como se aprecia en la definición de  $g$ , estamos hablando de una función compleja, y las representaciones mostradas en las figuras anteriores corresponden sólo a la parte real, si bien el filtro de Gabor realmente está compuesto de una parte real y una imaginaria que pueden verse como dos filtros iguales desfasados  $90^\circ$ .

La parte real contiene el filtro

$$g_r(x, y, \lambda, \theta, \phi) = e^{-\frac{x^2+y^2}{(\lambda/2)^2}} \cos\left(2\pi \frac{\theta}{\lambda}(x \cos \phi + y \sin \phi)\right) \quad (12)$$

La parte imaginaria contiene

$$g_i(x, y, \lambda, \theta, \phi) = e^{-\frac{x^2+y^2}{(\lambda/2)^2}} \sin\left(2\pi \frac{\theta}{\lambda}(x \cos \phi + y \sin \phi)\right) \quad (13)$$

Las respuestas de las componentes real e imaginaria de un filtro de Gabor complejo a una senoide son otra senoide, como puede apreciarse en la figura 4. Pero si lo que queremos es medir la presencia o energía de una señal de cierta frecuencia en una entrada, lo deseable es que la respuesta sea una señal positiva. Para ello, lo que haremos será tomar como salida del filtro Gabor la magnitud del mismo tomando parte real e imaginaria como coordenadas polares, es decir, la salida viene definida como la raíz cuadrada de la suma de los cuadrados de las salidas real e imaginaria en cada punto. A esto lo llamamos filtro de energía de Gabor y es el tipo que usamos en el presente proyecto con fines de identificación de texturas.

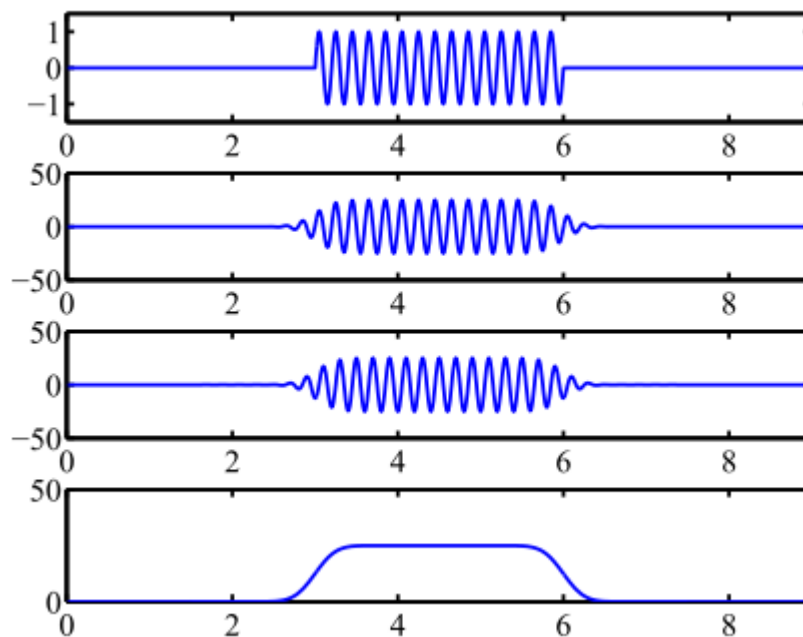


Figura 4: Filtro de energía de Gabor en 1-D

### 3.2.2. Implementación como matrices de convolución

Sobre la aplicación eficiente de estos filtros a una imagen, hay que contemplar el hecho de que en la teoría el filtrado resulta ser la convolución de la función de Gabor sobre la imagen completa para calcular la salida en cada punto, si bien esto es altamente costoso computacionalmente hablando.

Teniendo en cuenta que la salida del filtro depende de la región donde se centra su componente gaussiana, la aportación del resto de puntos de la imagen que quedan fuera de dicha campana va a ser prácticamente nula, por lo que en este proyecto se ha optado por modelar cada filtro de Gabor como una matriz cuadrada cuya dimensión depende de la escala del filtro, y por tanto de la desviación estándar de la componente gaussiana. En concreto, la dimensión  $N$  de dicho núcleo se calcula mediante la siguiente fórmula:

$$N = 2 \cdot \lambda + 1 \quad (14)$$

Una vez calculado un núcleo puede aplicarse como cualquier filtro de convolución lineal, es decir que su coste de computación es similar a aplicar por ejemplo un filtro de suavizado básico, si bien la dimensión de la matriz resulta ser mayor en el caso de los núcleos de los filtros de Gabor. Por ejemplo, para un filtro con  $\lambda=8$  la dimensión resulta  $N = 17$ .

De esta forma, en nuestro método tenemos para cada filtro Gabor dos matrices  $N \times N$ , una con la parte real y otra con la parte imaginaria del filtro, que se aplican por separado a cada canal de la imagen original por convolución 2-D, y sus resultados respectivos son interpretados como coordenadas polares de las que se extrae la magnitud correspondiente.

Esto provoca una diferencia respecto a la definición usual de filtros Gabor en la literatura, donde habitualmente se caracteriza a los mismos con sólo dos parámetros: frecuencia espacial y orientación, ligando la escala del filtro a la propia frecuencia. En este trabajo, sin embargo, hemos desligado ambos parámetros y por tanto necesitamos definir los tres para formar un filtro de Gabor como una matriz:

1. La escala  $\lambda$  de la que derivaremos el tamaño de dicha matriz de convolución según la fórmula (14), así como la desviación estándar de la componente gaussiana, que como se observa en las ecuaciones (10), (12) y (13) se define como  $\lambda/2$ .
2. La frecuencia  $\theta$  nos proporciona el número de ciclos de la senoide dentro de la escala  $\lambda$  indicada.
3. La orientación  $\varphi$  se mantiene como en la mayoría de las referencias bibliográficas existentes en la literatura.

### 3.3. Formación de vectores de características

Un banco de filtros de Gabor se construye especificando un conjunto de escalas  $\lambda_1 \dots \lambda_P$ , un conjunto de frecuencias espaciales  $\theta_1 \dots \theta_Q$ , y un conjunto de orientaciones  $\phi_1 \dots \phi_R$ , de manera que contendrá  $N=P \cdot Q \cdot R$  filtros con todas las posibles combinaciones de parámetros.

Tras la aplicación del banco de filtros de Gabor a los canales originales de la imagen, y a partir de la información contenida en las matrices resultantes, se procede a describir la textura presente en el entorno de cada píxel de la imagen original, mediante un vector de características que tendrá tantos valores como filtros se hayan aplicado multiplicado por el número de canales originales, que en nuestro caso son dos (canales  $a$  y  $b$  en CIELab).

Sea  $G_{i,j,k}^a$  la matriz resultante de la aplicación del filtro con parámetros  $\lambda_i, \theta_j, \phi_k$  sobre el canal  $a$  de la imagen original, y de forma similar  $G_{i,j,k}^b$  la salida del mismo filtro para el canal  $b$ . El vector de características asociado al entorno del píxel  $(x,y)$  en la imagen original se compone de la siguiente forma:

$$V(x, y) = \begin{bmatrix} G_{1,1,1}^a(x, y) \\ \dots \\ G_{1,1,R}^a(x, y) \\ \dots \\ G_{1,Q,R}^a(x, y) \\ \dots \\ G_{P,Q,R}^a(x, y) \\ G_{1,1,1}^b(x, y) \\ \dots \\ G_{1,1,R}^b(x, y) \\ \dots \\ G_{1,Q,R}^b(x, y) \\ \dots \\ G_{P,Q,R}^b(x, y) \end{bmatrix}$$

Realmente, aunque el mostrado es el orden que se utiliza en la implementación del método empleada en los experimentos, el orden concreto de las características en el vector no es relevante siempre que se mantenga el mismo en todos los vectores generados. La idea importante es que el vector de características del píxel  $(x,y)$  recoge el valor en dichas coordenadas  $(x,y)$  de cada matriz resultante de la aplicación de cada filtro Gabor, en un orden fijo.

### 3.4. Agrupamiento mediante K-Means

Una vez descrita la textura que rodea cada píxel de la imagen mediante un vector de características, queda la tarea de clasificar los mismos en grupos que correspondan a las diferentes texturas buscadas o presentes en la imagen.

Nótese aquí que el número de texturas buscadas, o texturas de interés para la aplicación, normalmente va a ser mucho menor que el número de texturas realmente presentes en la imagen, especialmente cuando estamos trabajando con imágenes reales. En concreto, en este trabajo las texturas de interés son las vegetales, si bien las imágenes disponibles contienen otros muchos elementos, algunos esperables como son tierra y cielo, y otros muy difíciles de prever, como casas, puentes, torres eléctricas, y en general elementos de índole ajena al tema de estudio.

Existen numerosos algoritmos de agrupamiento, de varios tipos, siendo el elegido finalmente en el presente proyecto por su sencillez y eficiencia el algoritmo K-Means [7] [17] [18]. Se trata de un algoritmo de agrupamiento no supervisado que representa una distribución de datos usando K centros, donde K es un número elegido por el usuario.

El algoritmo K-Means tiende a encontrar grupos naturales en los datos. El usuario elige el número de grupos a identificar y el algoritmo crea K centros que acaba situando, dentro del espacio de características, en el centro de los grupos naturales de datos existentes en el mismo.

El funcionamiento básico del algoritmo es el siguiente:

1. Como entrada se recibe el conjunto de datos y el número K de grupos deseados.
2. Se colocan aleatoriamente K centros en el espacio de datos.
3. Se asocia a cada dato el centro más cercano al mismo.
4. Se calcula, para cada centro, el punto medio de todos los datos asociados al mismo, y se produce un acercamiento del centro a dicho punto medio.
5. Se vuelve al punto 3 hasta que haya convergencia, esto es, los centros no se muevan de forma relevante.



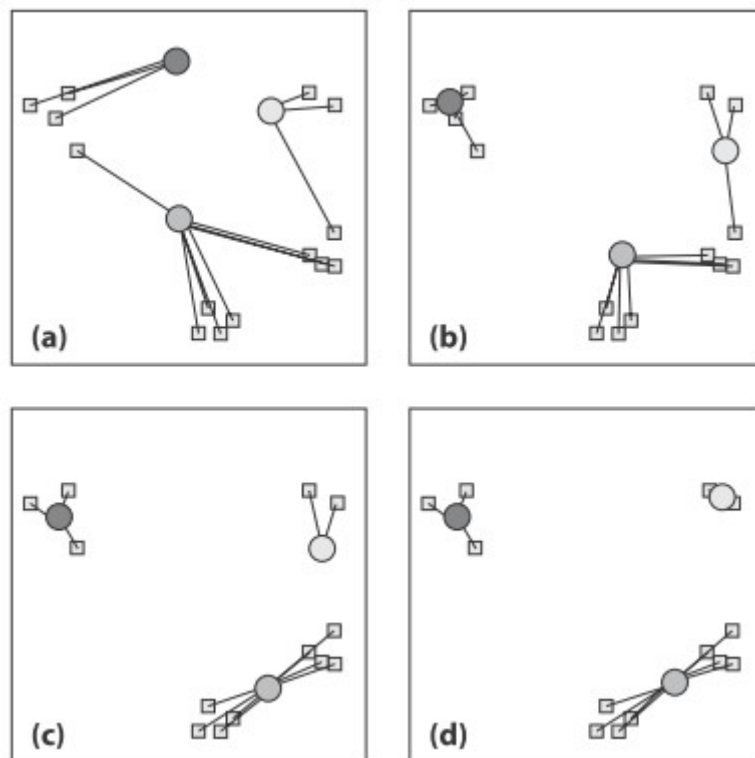


Figura 5: Convergencia de los centros en algoritmo K-Means

En la mayoría de los casos el algoritmo converge rápidamente, aunque a veces puede requerir un número mayor de iteraciones.

Por lo general se trata de un algoritmo muy efectivo, pero tiene algunos problemas [17] [18], principalmente:

1. No se garantiza una solución óptima global: puede que se caiga en soluciones localmente óptimas.
2. El número de grupos elegido depende del usuario. Si por ejemplo en la figura 5 se hubieran elegido sólo dos grupos los resultados serían muy diferentes, probablemente poco intuitivos.

Por suerte, para cada uno de estos problemas existe alguna solución, o al menos alguna forma de reducirlo. Para ello, se puede usar como medida de la calidad de la clasificación la varianza en los datos de cada grupo clasificado, o alguna otra medida de dispersión, de forma que un mejor agrupamiento es aquel que la minimiza sin usar demasiados grupos.



Las ~~posibles~~ soluciones a los problemas anteriores son respectivamente:

1. Ejecutar el algoritmo varias veces con centros iniciales diferentes, y elegir la solución con la menor varianza.
2. Comenzar ejecutando el algoritmo con  $K=1$ , y volver a ejecutar iterativamente incrementando  $K$ , midiendo la varianza total para cada iteración, de forma que ~~esta~~ vaya disminuyendo hasta llegar a un punto de estancamiento, momento en el cual el nuevo grupo añadido no reduce de forma relevante la varianza.

En especial, el identificado en el segundo punto es un problema que nos hemos encontrado en los experimentos realizados, y que veremos con más detalle en la siguiente sección.



#### 4. Experimentaciones realizadas y resultados observados

Desde el principio el camino a explorar fue la aplicación de filtros de Gabor para clasificar cada píxel en función a su entorno. Es decir, producir una descripción, en forma de vector de características, de cada píxel en función del resultado obtenido de la aplicación de un banco de filtros de Gabor en su entorno, y utilizar este espacio de características como entrada de algún algoritmo de agrupamiento para realizar la clasificación.

Debido a que las imágenes tratadas son en color, una primera batería de experimentos consiste en estudiar la importancia de los diferentes espacios de color, confirmando finalmente la importancia de la elección apropiada del mismo.

En la figura 6, se representa el esquema simplificado del proceso realizado. Dada la imagen de entrada en el modelo de color RGB, ésta se transforma a un espacio de color uniforme. Seguidamente se realiza un filtrado de Gabor para obtener un vector de características asociado a cada píxel. Con dichos vectores se realiza un agrupamiento para distribuirlos en clases, de acuerdo con el método K-Means propuesto.

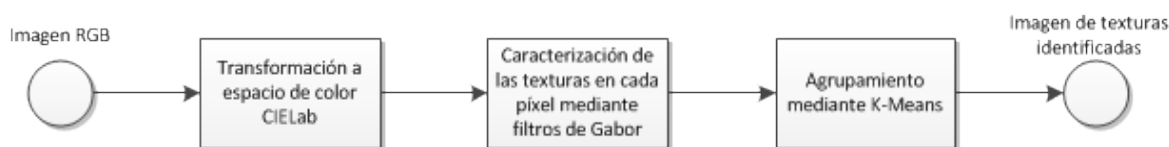


Figura 6: Idea básica inicial del método

Los resultados obtenidos con el espacio RGB utilizado inicialmente no eran nada alentadores, incluso empeorando si se pretrataba la imagen normalizando los valores para pasar al espacio cromático rgb, tratando de independizarnos de los efectos de la luminosidad en la imagen. Se probaron distintas configuraciones de parámetros para el banco de filtros de Gabor aplicado, no encontrando una mejora general con ninguna combinación de escalas y orientaciones, o dicho de otra forma, no encontrando el espacio de características adecuado.

Se pasó entonces a probar sobre el espacio HSL, mejorando algo los resultados de forma que las clasificaciones resultantes de aplicar los mismos filtros de Gabor, es decir, para un mismo espacio de características, eran más parecidas a las esperadas.

No obstante, la mejora más notable vino con la utilización del espacio CIE Lab como base para la aplicación del banco de filtros, en concreto de sus componentes cromáticas "a" y "b", obviando el canal de intensidad o luminosidad "L". Esto dio lugar a una clasificación mucho más cercana a la esperada, es decir resultante en la tendencia al agrupamiento en una misma clase de píxeles

pertenecientes al mismo tipo de elementos naturales, de la forma en que lo haría una persona basada en su capacidad perceptual. Además, es esperable que el hecho de descartar el canal de luminosidad implique cierta independencia respecto de las condiciones climáticas y de iluminación en las que se tomen las imágenes.

De todas formas, el grueso de pruebas y experimentación se ha dedicado a realizar [ejecuciones](#) con diferentes parámetros en el diseño del banco de filtros de Gabor. Se han hecho pruebas con bancos de filtros con una sola y con varias escalas, con número diferente de orientaciones, y con varias frecuencias espaciales.

Destacar también que para afinar los resultados se experimentó con el suavizado [tanto](#) de la imagen original como de las resultantes de la aplicación de los filtros de Gabor, observando que en ambos casos la clasificación mejoraba al aplicar un filtro de suavizado gaussiano sobre las mismas, haciendo menos frecuente la aparición de pequeñas "islas" de píxeles mal clasificados, y aumentando la tendencia a que píxeles próximos se clasificasen como de una misma textura. Es de suponer que el filtrado de la imagen inicial procura este efecto al limar posibles defectos de la misma debido al ruido, de hecho el ruido por la compresión [jpg](#) en las imágenes originales era observable a simple vista. En cuanto al filtrado de las resultantes de los filtros, [es](#) una técnica recomendada en la literatura consultada, y su aplicación ha servido para amortiguar la variabilidad presente en las texturas naturales, como es nuestro caso.

Por último, también se experimentó normalizando las salidas de los filtros, puesto que los rangos de los valores obtenidos eran muy diferentes en cada característica, es decir, en la resultante de cada filtro de Gabor. Algunos filtros, en general los de mayor escala, producen un rango de valores más amplio a la salida, lo que significa que las variaciones en distancia entre estas características resultan ser mucho mayores que en el resto, influyendo por tanto mucho más en la clasificación de la textura. En principio, no es lo deseable, y de hecho en las pruebas realizadas se ha comprobado empíricamente que los resultados son mejores aplicando esta normalización, que sirve para forzar que todos los filtros de Gabor tengan el mismo rango de salida.

En definitiva, tras el conjunto de pruebas experimentales, el proceso general se diseña tal y como se describe en el diagrama de flujo de la figura 7.

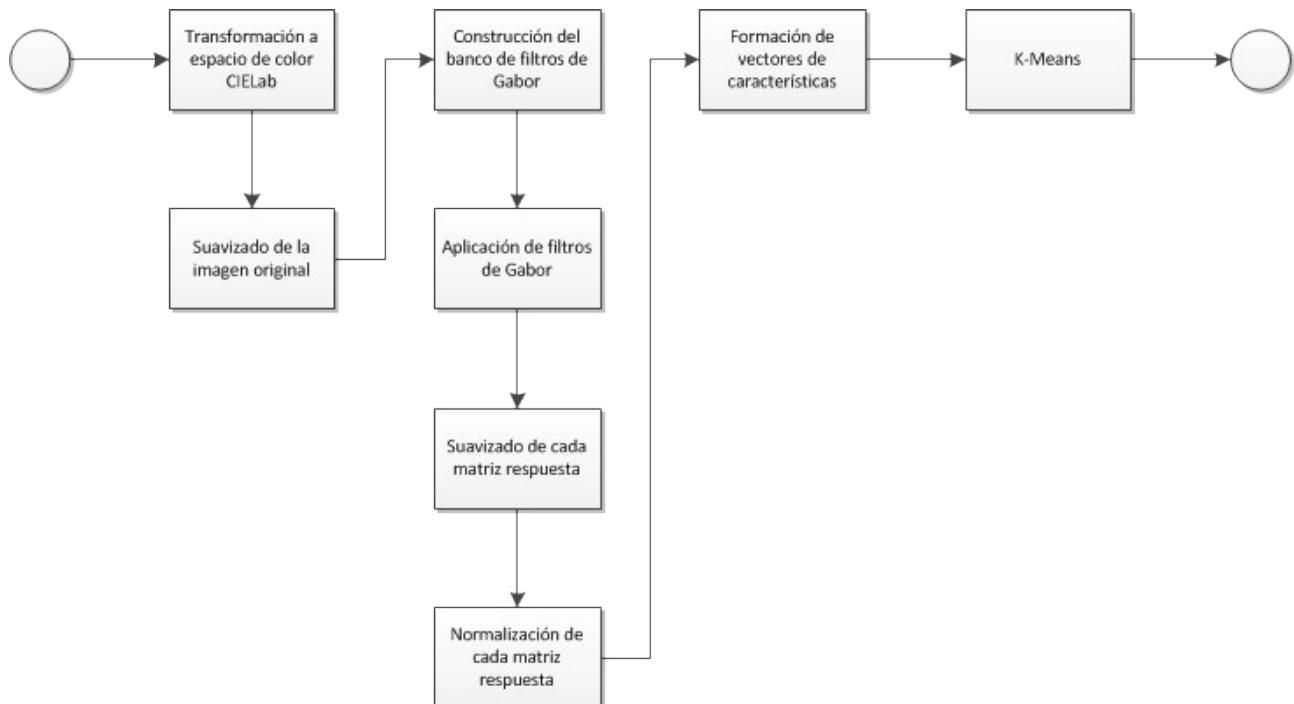


Figura 7: Descripción gráfica del método propuesto



#### 4.1. Ejemplo detallado de clasificación

Se expone a continuación un ejemplo de aplicación del método a la imagen concreta mostrada en la figura 8, elegida como ejemplo del banco de imágenes disponible, formado por alrededor de 30 de ellas. Los resultados y conclusiones son similares a los obtenidos en el resto de imágenes analizadas en los experimentos llevados a cabo en el presente proyecto.



Figura 8: Imagen original para ejemplo de aplicación

La primera etapa del proceso es convertir la imagen al espacio de color CIELab. En la figura 9 se muestra el canal  $L$  correspondiente a la información de luminosidad. En las figuras 10 y 11 se muestran, respectivamente, los canales  $a$  y  $b$  de cromaticidad de la imagen.



*Figura 9: Canal  $L$  en CIELab*



*Figura 10: Canal  $a$  en CIELab*



*Figura 11: Canal  $b$  en CIELab*

De los canales obtenidos, nótese cómo el canal de luminosidad exhibe mucho mayor contraste que los de cromaticidad, que describen el color. Esto encaja con la teoría de que la luminosidad tiene mucha más relevancia, en la percepción visual humana, que la cromaticidad. En cualquier caso, en el presente proyecto la luminosidad supone una fuente de variabilidad de la que intentamos desprendernos, por lo que se descarta el canal  $L$  pasando a trabajar únicamente con la cromática de la imagen, canales  $a$  y  $b$ , que aunque en las figuras muestran un bajo contraste, como veremos contienen la información necesaria para poder analizar las diferentes texturas.

El siguiente paso en el proceso consiste en generar el banco de filtros de Gabor que se aplicará sobre dichos canales. El diseño de este banco de filtros se ha obtenido de forma completamente empírica, jugando con los parámetros del mismo y comparando los resultados obtenidos. De los experimentos llevados a cabo las clasificaciones más convincentes se han conseguido con un banco compuesto por 24 filtros en 3 escalas (4, 8 y 16 píxeles), 4 orientaciones (horizontal, vertical y diagonales) y 2 frecuencias espaciales (1 y 1.5).

El significado de estos parámetros es el ya expuesto al final de la sección 3.2.2. En las figuras 12 a 19 se muestran los diferentes filtros generados, ampliados y representados en el plano de una imagen. Se obvia en las representaciones el parámetro de escala, pues la única diferencia es el tamaño o dimensión del núcleo generado, y aquí se muestran ampliados para facilitar su observación. Obsérvese como influyen la orientación  $\varphi$ , en la dirección del filtro, y la frecuencia espacial  $\theta$ , en el número de ciclos de la sinusoide.

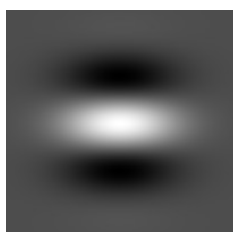


Figura 12: Filtro Gabor  $\varphi=0$  y  $\theta=1$

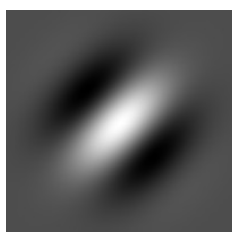


Figura 13: Filtro Gabor  $\varphi=\pi/4$  y  $\theta=1$

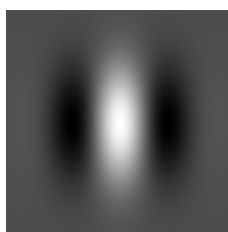


Figura 14: Filtro Gabor  $\varphi=\pi/2$  y  $\theta=1$

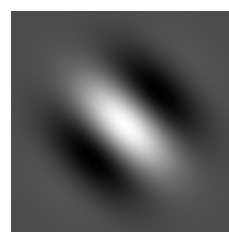


Figura 15: Filtro Gabor  $\varphi=3\pi/4$  y  $\theta=1$

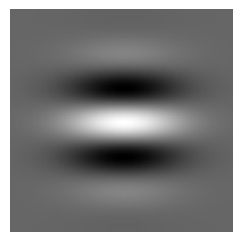


Figura 16: Filtro Gabor  $\varphi=0$  y  $\theta=1.5$

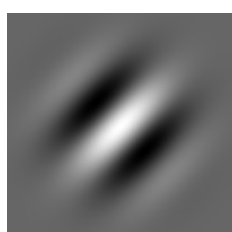


Figura 17: Filtro Gabor  $\varphi=\pi/4$  y  $\theta=1.5$

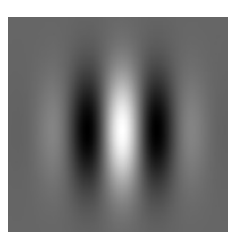


Figura 18: Filtro Gabor  $\varphi=\pi/2$  y  $\theta=1.5$

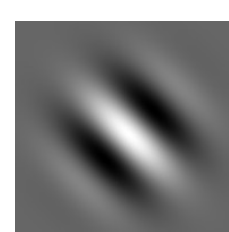


Figura 19: Filtro Gabor  $\varphi=3\pi/4$  y  $\theta=1.5$

Realmente, la representación mostrada en las figuras 12 a 19 corresponde a la parte real del filtro de Gabor complejo. La representación de la parte imaginaria sería idéntica pero con las tonalidades de gris invertidas, pues están en fase de  $90^\circ$ .

De la aplicación de este banco de 24 filtros sobre cada uno de los 2 canales considerados de la imagen original,  $a$  y  $b$ , resultan 48 matrices que contienen las respuestas de los filtros a cada canal. Dichas matrices son suavizadas mediante un filtro gaussiano, y sus valores normalizados, según se explicó en la descripción del método, al principio de esta sección. En las figuras 20 a 23 se muestran algunas de ellas, es decir, las respuestas de varios filtros de Gabor con distintos parámetros, especificados al pie de cada figura.

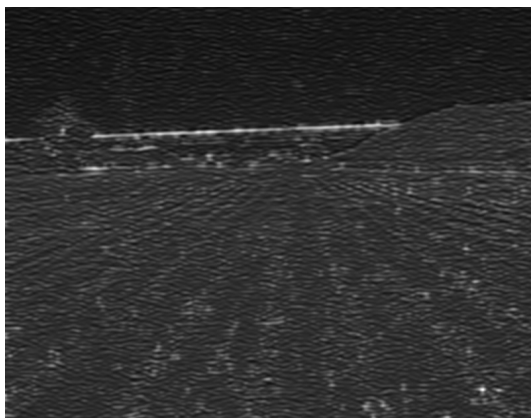


Figura 20: Respuesta de un filtro de Gabor con  $\lambda=8$ ,  $\theta=1.5$ ,  $\varphi=0$ , sobre canal  $b$

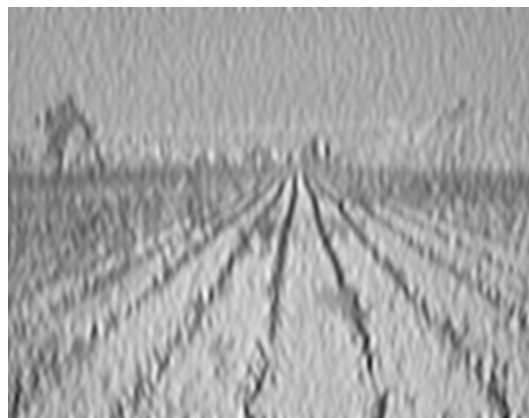


Figura 21: Respuesta de un filtro de Gabor con  $\lambda=16$ ,  $\theta=1.0$ ,  $\varphi=\pi/2$ , sobre canal  $a$

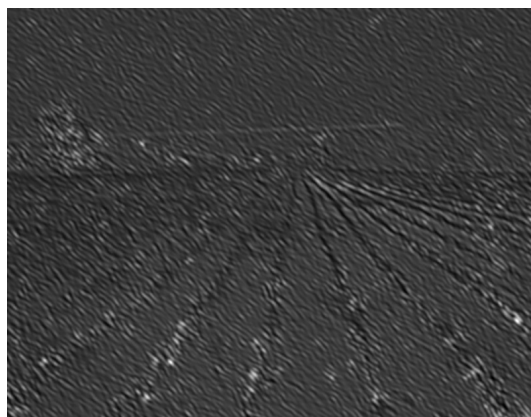


Figura 22: Respuesta de un filtro de Gabor con  $\lambda=8$ ,  $\theta=1.5$ ,  $\varphi=3\pi/4$ , sobre canal  $a$



Figura 23: Respuesta de un filtro de Gabor con  $\lambda=4$ ,  $\theta=1.0$ ,  $\varphi=\pi/4$ , sobre canal  $b$

Obsérvese cómo se realzan los elementos de la imagen que tienen bordes en la misma orientación que el filtro aplicado, por ejemplo el puente en la figura 20, a la que se ha aplicado el filtro horizontal de la figura 16, los surcos verticales en la figura 21 por haber aplicado el filtro vertical de la figura 14, y los surcos diagonales a la derecha en la figura 22, donde el filtro aplicado fue el de la figura 19.



Reorganizando la información de dichas 48 matrices, tal como se explica en la sección 3.3, se forma un vector de características de dimensión 48 por cada píxel de la imagen original. Dichos vectores corresponden a la descripción de la textura en el entorno centrado en el píxel correspondiente.

Una vez tenemos descrito cada píxel de la imagen en cuanto a la textura en su entorno local, el siguiente paso, de acuerdo con el esquema de las figuras 6 y 7, es el agrupamiento. Como se explicó en la sección anterior, se utiliza el algoritmo K-Means eligiendo el número de clases en las que se pretende ubicar cada píxel de la imagen original. Veremos aquí el resultado de diferentes ejecuciones del K-Means sobre el mismo conjunto de vectores de características ya obtenido para la imagen original de la figura 8, variando el número de grupos elegido, y analizaremos el segundo de los problemas expuesto en la sección 3.4.

En primera instancia, probamos a agrupar en tres clases, que intuitivamente esperábamos que resultaran ser cielo, tierra y estructuras vegetales. En la figura 24 se muestra la clasificación resultante, que cumple de forma bastante aproximada con dicha expectativa, aunque no con la precisión deseable puesto que yerra clasificando algunos píxeles que en la realidad son pertenecientes a estructuras vegetales, y deberían etiquetarse en color verde, en el mismo grupo que los píxeles de tierra, en color azul. Para una mejor observación de los resultados, se superpone también la clasificación coloreada a la imagen original, ~~en la figura 25.~~

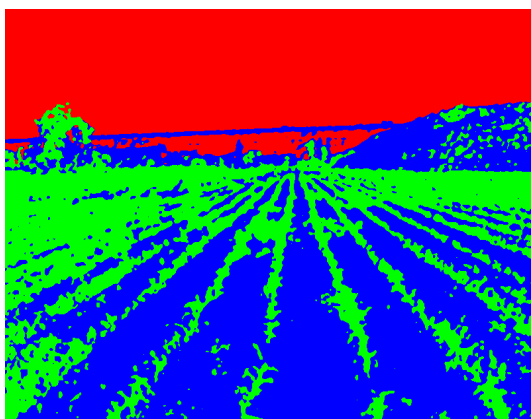


Figura 24: Agrupamiento en 3 clases



Figura 25: Superposición del agrupamiento en 3 clases a la imagen original

No obstante, el objetivo del presente proyecto no es sólo clasificar en estos tres grupos, sino llegar a distinguir diferentes tipos de estructuras vegetales presentes en las imágenes, para lo que es necesario por tanto utilizar más grupos en la clasificación, que ~~es necesario asignar~~ al algoritmo de clasificación K-Means.



Al aumentar el número de clases a cuatro, como se observa en las figuras 26 y 27, efectivamente la clase adicional sirve para distinguir entre dos tipos de estructuras vegetales: una agrupa los surcos principales correspondientes a los cultivos, con un pequeño error pues incluye algunos píxeles correspondientes a vegetación arbórea; y otra que agrupa la vegetación circundante a los surcos cultivados, los arbustos de la colina en el fondo de la imagen a la derecha, y la gran mayoría de la vegetación arbórea. Es decir, la clasificación se aproxima bastante a una identificación, dentro de la primera clase, de las estructuras vegetales correspondientes a las plantas cultivadas.

Nótese, observando la imagen original y las figuras 26 y 27, que **la distinción entre las clases no depende sólo de la tonalidad del verde, sino que obedece bastante al tipo de vegetación en sí**, o al menos a diferentes estados de una misma planta, por ejemplo de crecimiento, aunque con los errores mencionados, que seguramente podrían corregirse perfeccionando el diseño del banco de filtros. Esto es una **aportación fundamental de la aplicación de filtros de Gabor** frente al análisis de la tonalidad de cada píxel de manera individual, que como por ejemplo en [1] distinguen varios tipos de planta donde en realidad hay píxeles con diferentes tonalidades de verde pertenecientes a una misma planta, es decir, producen una sobreclasificación.

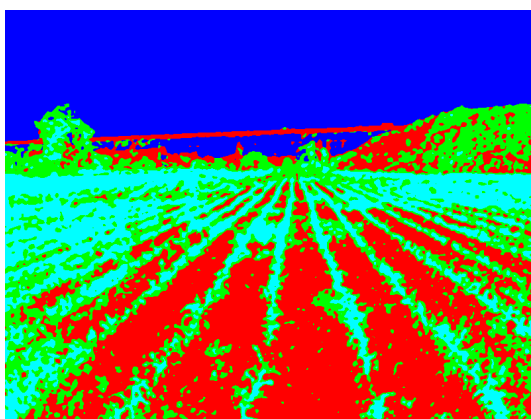


Figura 26: Agrupamiento en 4 clases



Figura 27: Superposición del agrupamiento en 4 clases a la imagen original

No obstante, que el objetivo sea identificar cielo, tierra, plantas cultivadas y otras plantas no quiere decir que en la imagen no existan más texturas, por lo que si forzamos a que la clasificación se haga en cuatro grupos estamos obligando a clasificar erróneamente todas las texturas que no pertenezcan realmente a ninguno de dichos grupos. Por ello, se hicieron pruebas utilizando 6 y 8 grupos de clasificación respectivamente, obteniendo los resultados que se visualizan en las figuras 28 a 31.

En la agrupación en 6 clases, figuras 28 y 29, puede observarse cómo uno de los dos nuevos grupos ha servido para distinguir entre dos texturas diferentes presentes en lo que un humano clasificaría como cielo, y el otro se corresponde con zonas que un humano describiría como suelo con algo de vegetación.

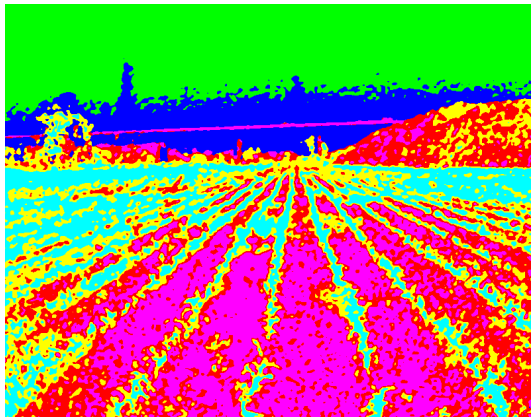


Figura 28: Agrupamiento en 6 clases



Figura 29: Superposición del agrupamiento en 6 clases a la imagen original

Se siguen teniendo de todas formas clasificaciones ambiguas debido a que el número de texturas en la imagen es mayor que el número de grupos empleados en el método K-Means. Por ejemplo, el puente aparece clasificado en el mismo grupo que el suelo, cuando en realidad son dos elementos o texturas que nada tienen que ver entre sí y que serían clasificados separadamente por un humano. Este caso concreto no tiene mucho interés para nuestra aplicación porque lo que realmente nos interesa es la vegetación, pero de forma similar no se está separando bien la vegetación arbórea, que muestra más variabilidad que la vegetación en la zona de cultivo, y se está mezclando aquella, por partes, con los distintos grupos de esta.

Sin embargo, si seguimos aumentando el número de clases en el agrupamiento, observamos a continuación cómo la precisión mejora, al aproximarse el número de grupos al número real de elementos o texturas diferentes en la imagen.

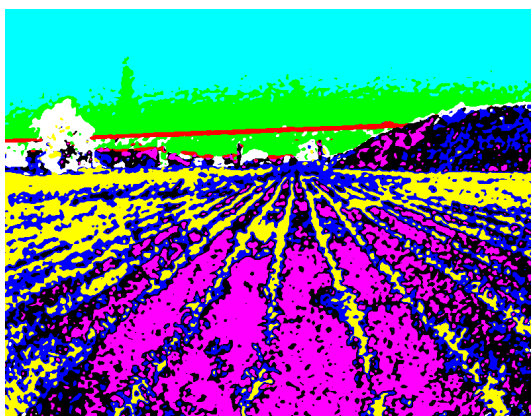


Figura 30: Agrupamiento en 8 clases



Figura 31: Superposición del agrupamiento en 8 clases a la imagen original

En los resultados obtenidos considerando ocho clases ya se aprecia un nivel de precisión aceptable en cuanto a la homogeneidad de cada grupo clasificado, de forma que aunque no todos los grupos sean igual de relevantes de cara al objetivo del proyecto, al aumentar el número de estos se han conseguido principalmente dos cosas: poder descartar grupos sin interés con mayor fiabilidad, y disminuir el número de píxeles erróneos en los grupos relevantes, que ya no se ven tan "obligados" a absorber píxeles "huérfanos", entendiendo por tal a un píxel que pertenece a una textura real que no tiene correspondencia con ninguna de las clases contempladas por K-Means, aunque aún sigue habiendo algunos, como las columnas del puente, que se han absorbido por uno de los grupos de suelo, lo que en principio y según lo expuesto se arreglaría si continuamos aumentando el número de clases a discernir.

Para ilustrar lo expuesto con una mayor claridad, en las figuras 32 a 39 se muestran las ocho clases identificadas sobre la imagen original de forma aislada, nombrando descriptivamente cada una de ellas.

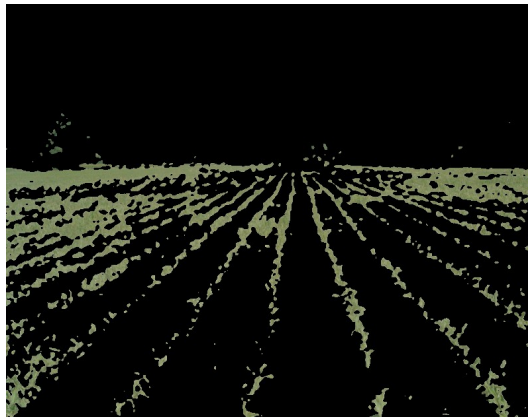
1. *Tierra*: se agrupan los píxeles correspondientes a suelo sin apenas trazas de vegetación.



*Figura 32: Píxeles clasificados como terreno sin vegetación*

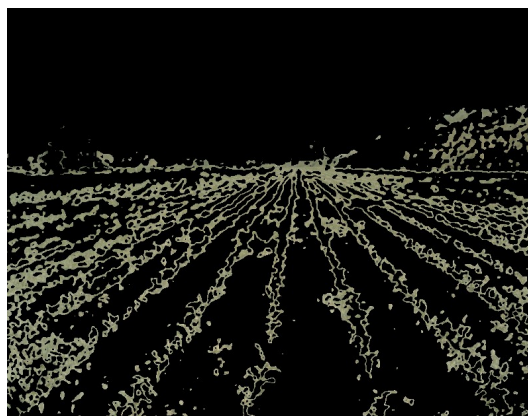


2. *Cultivos*: agrupa los píxeles que un humano clasificaría como plantas cultivadas. Véase que aún se observan algunos píxeles que realmente pertenecen a un árbol, mal incluidos en este grupo. Es de suponer que un perfeccionamiento del banco de filtros o un algoritmo de agrupamiento más sofisticado podrían disminuir la ocurrencia de este tipo de errores.



*Figura 33: Píxeles clasificados como cultivo*

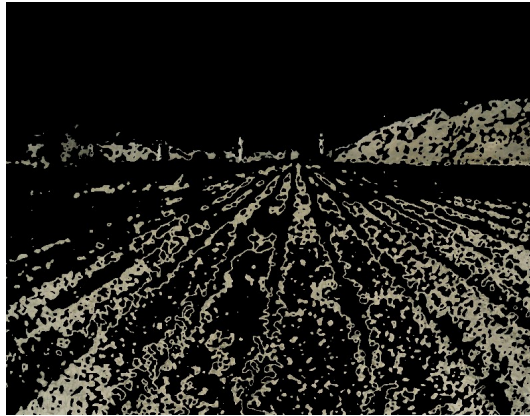
3. *Pastos*: Esta clase agrupa píxeles que un observador catalogaría como vegetación similar a los cultivos pero no correspondiente a los mismos, como por ejemplo pastos y hierbas altas, anexas a aquellos. Se puede observar cómo incluye tanto la vegetación que rodea al maíz como los pastos en la colina del fondo.



*Figura 34: Píxeles clasificados como pastos*

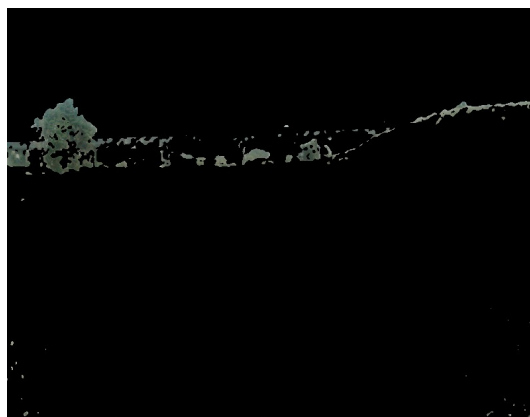


4. *Hierbas*: Incluye los píxeles correspondientes al suelo donde existe cierta vegetación baja que matiza la textura del mismo, como hierbas. Se incluyen erróneamente en esta clase las columnas del puente, debido a que no existe una clase exclusiva para ellos y a que tienen una textura muy parecida debido a la resolución de la imagen.



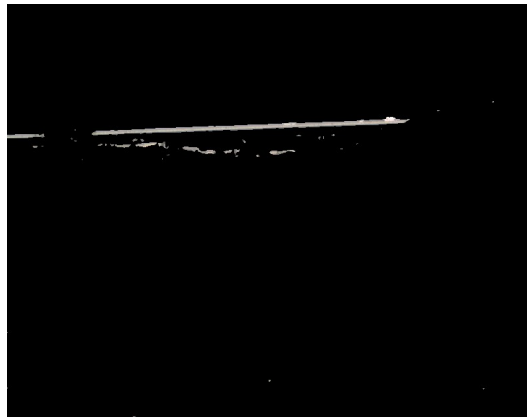
*Figura 35: Píxeles clasificados como hierbas*

5. *Arboleda*: incluye los píxeles correspondientes a los árboles y arbustos presentes en la imagen. Aparecen clasificados aquí erróneamente algunos píxeles del puente con una tonalidad similar. Probablemente una imagen de mejor resolución, un diseño más óptimo del banco de filtros y el uso de más clases en el agrupamiento son factores que ayudarían a corregir este tipo de errores. No obstante la aproximación como se ve en la imagen es aceptable.



*Figura 36: Píxeles clasificados como arboleda*

6. *Sin interés tipo I* : incluye píxeles que corresponden a la parte superior del puente y de las construcciones que sobresalen de entre la vegetación en el horizonte, similares en cuanto a la textura observada, pero sin interés desde el punto de vista de clasificación de texturas en imágenes agrícolas.



*Figura 37: Píxeles sin interés, tipo I*

7. *Sin interés tipo II*: Agrupa los píxeles que corresponden a la parte más baja del cielo, cuya textura se ve matizada por la presencia de una serie de torres y tendido eléctrico.



*Figura 38: Píxeles sin interés, tipo II*

8. *Sin interés tipo III*: píxeles correspondientes a cielo abierto.



*Figura 39: Píxeles sin interés, tipo III*

El hecho de tener estos tres últimos grupos es de relevancia no por el interés de los mismos, sino porque sirven para eliminar dichos píxeles de los grupos donde se clasificaban erróneamente al usar un número menor de clases. Una vez que se tienen aislados, pueden simplemente descartarse, ~~y los grupos de interés son~~ realmente más precisos y homogéneos.



## 4.2. Otros ejemplos

Se muestran a continuación los resultados de la aplicación del método a otras imágenes.

En la figura 40 se muestra la imagen original, en la que se aprecia que la iluminación es baja. La figura 41 muestra un mapa de colores que representa las diferentes clases identificadas en la imagen original. La clase representada en color blanco corresponde a los píxeles clasificados como cultivo, que se representan también aislados en la figura 42. La clase representada en color azul oscuro puede interpretarse como píxeles identificados como otra vegetación, incluyendo las plantas y hierbas entre los surcos y los árboles del horizonte. Se representa aislada en la figura 43. Nótese los píxeles erróneamente incluidos que realmente pertenecen a nubes.



Figura 40: Imagen con poca iluminación

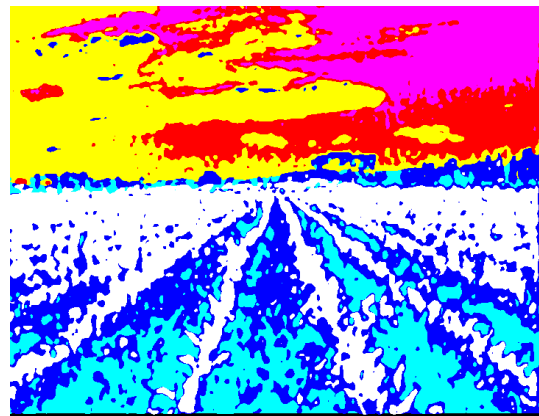


Figura 41: Clasificación resultante

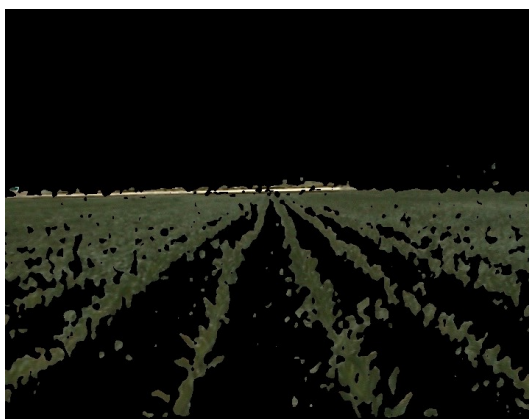


Figura 42: Surcos cultivados



Figura 43: Otra vegetación



Otra imagen diferente es la mostrada en la figura 44, donde se observan unas condiciones de iluminación menos homogéneas, al estar la mayoría del terreno en sombra excepto un parche del terreno donde el sol incide directamente, cambiando la tonalidad de las plantas. Las diferentes clases identificadas se muestran en un código de colores en la figura 45. La clase representada en azul oscuro corresponde a las plantas cultivadas. Esta clase se muestra aislada en la figura 46, apreciándose cómo incluye tanto la parte soleada como la más oscura, confirmandose que el método tolera condiciones de iluminación no óptimas, y que es capaz de agrupar texturas similares aunque con diferentes tonalidades de color en la misma clase. La clase representada en color magenta en la figura 45 corresponde a píxeles de otro tipo de vegetación. Se muestra esta clase aislada en la figura 47, donde se observa que la clase incluye tanto las hierbas y plantas entre surcos como la arboleda de fondo. También se vuelven a apreciar píxeles mal clasificados de algunas nubes.



Figura 44: Imagen con iluminación no uniforme

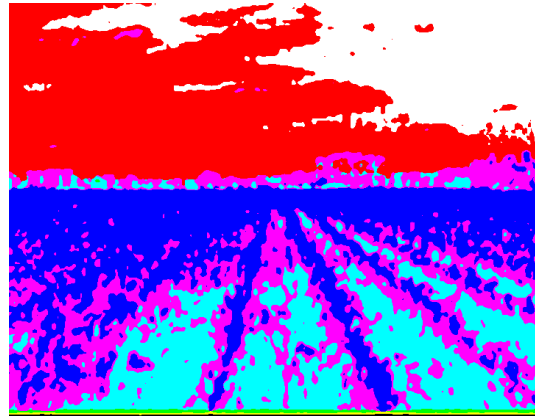


Figura 45: Clasificación resultante

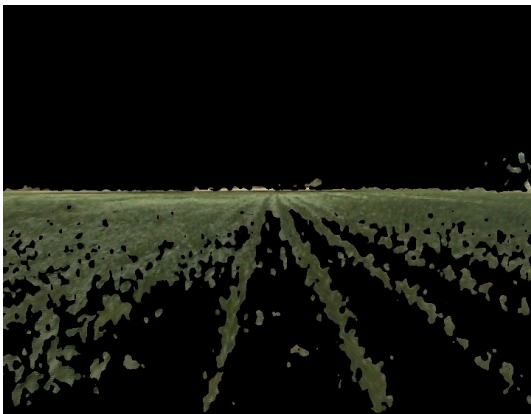


Figura 46: Surcos cultivados



Figura 47: Otra vegetación

#### 4.3. Comparación de resultados respecto al método de referencia

Se presenta en esta sección una comparación de los resultados del método propuesto con respecto al método de referencia [1], sobre la misma imagen original, mostrada en la figura 48. Ponemos el foco en el reconocimiento de texturas vegetales, y en la diferenciación de diferentes tipos de estas.

Los resultados de clasificación del método de referencia [1] se presentan en la figura 49. En ella, se representan coloreados en azul, rojo y amarillo los píxeles que han sido identificados como vegetación, y el resto muestran su aspecto original. Los tres colores mencionados representan las tres clases de vegetación identificadas por el método.



Figura 48: Imagen original de referencia

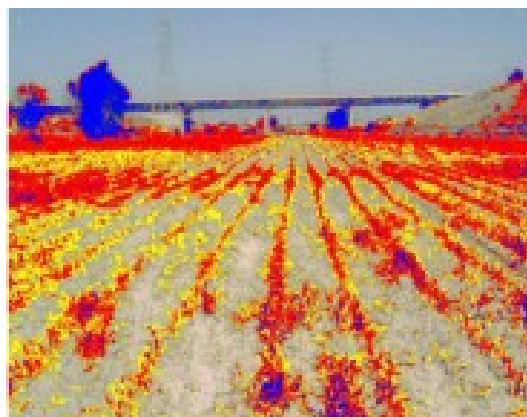


Figura 49: Clasificación de referencia.

Aunque la clasificación es bastante buena en general, respecto a la identificación de vegetación, se observan las siguientes imprecisiones:

1. Existe cierta vegetación débil que no ha sido clasificada como tal, sino que se ha clasificado como un tipo de suelo, como son las hierbas presentes entre los surcos, y parte de los pastos en la colina del fondo. Si bien dicha vegetación es tan escasa que quizá esto carezca de importancia.
2. Se ha clasificado como vegetación parte del puente.

Con respecto a la diferenciación de varios tipos de vegetación, se observa que la diferencia entre los tres grupos identificados es la tonalidad del verde, pero esto parece no corresponder completamente a diferentes tipos de vegetación, si consideramos las siguientes observaciones:

1. Se clasifican en el mismo grupo, representados en color azul en la figura 49, los píxeles pertenecientes a árboles, y un número de píxeles situados en los propios surcos cultivados.

2. Se clasifican en el mismo grupo, representado en color rojo en la figura 49, los píxeles pertenecientes a pastos en la colina a la derecha, y gran parte de los píxeles que pertenecen a cultivo.

Por otra parte, las plantas que pertenecen a cultivo, se observa que están repartidas, más o menos en la misma cantidad de píxeles, entre las dos clases representadas en rojo y amarillo en la figura 49, lo que sí parece obedecer a un diferente estado de crecimiento de las mismas, que muestran diferentes tonalidades de verde.

En cuanto a los resultados de clasificación del método propuesto en este proyecto, se presentan en las figuras 50 a 56. En la figura 50, se representa mediante un mapa de colores la clasificación obtenida mediante el método propuesto, utilizando ocho grupos en el K-Means. En la figura 51 se representan aislados los píxeles correspondientes a la clase representada en magenta en la figura 50, que se observa que corresponden a vegetación cultivada en avanzado estado de crecimiento, aunque se clasifican incorrectamente algunos píxeles del árbol, a la izquierda. En la figura 52 se representa aislada la clase representada en negro en la figura 50, que observamos se compone de cultivos en un estado de crecimiento más temprano que el anterior. También aquí se incluyen por error algunos píxeles del mismo árbol, y de algunos pastos altos.

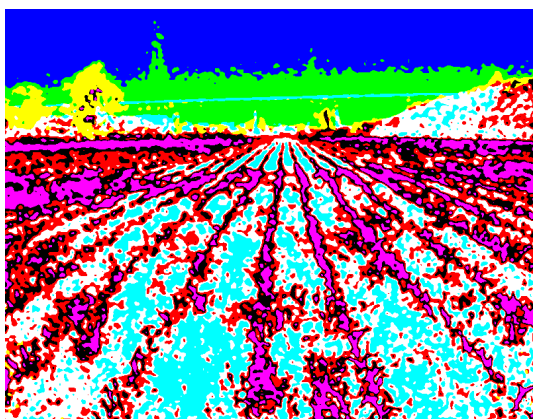


Figura 50: Clasificación en ocho grupos

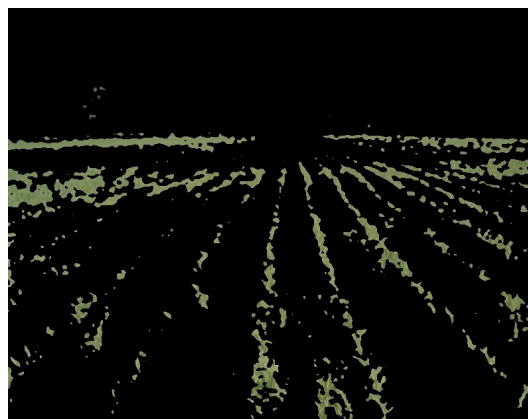


Figura 51: Vegetación tipo I

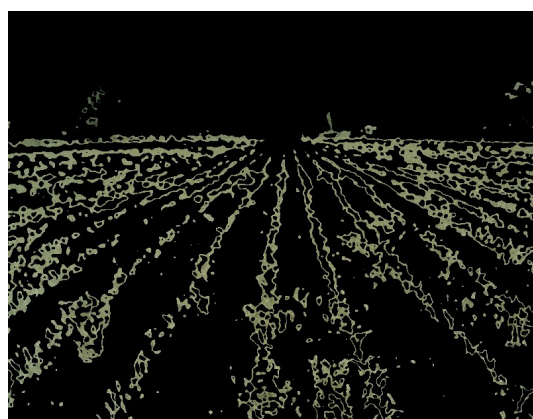


Figura 52: Vegetación tipo II

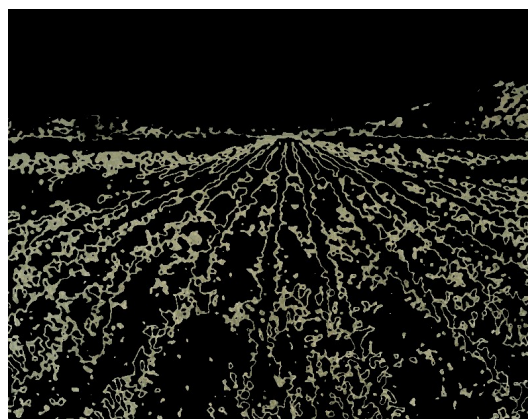
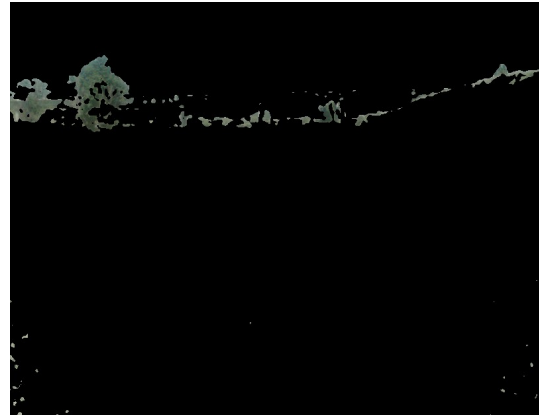


Figura 53: Vegetación tipo III

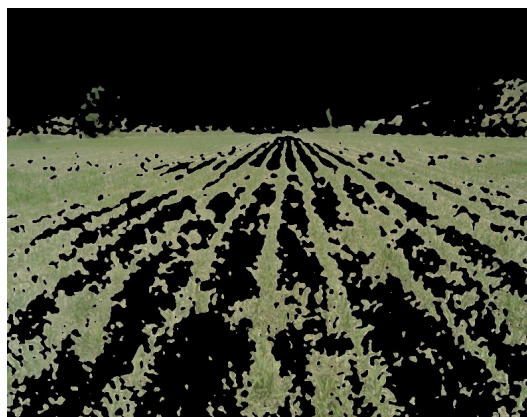


*Figura 54: Vegetación tipo IV*



*Figura 55: Vegetación tipo V*

La figura 53 muestra la clase correspondiente al color rojo en la figura 50, apreciándose que incluye cultivos poco crecidos, hierbas altas y pastos frondosos, que son difíciles de distinguir en la imagen. La figura 54 responde a la clase etiquetada en blanco en la figura 50, y corresponde como se aprecia a hierbas y pastos menos espesos. En la figura 55 está aislada la clase representada en amarillo en la figura 50, que contiene casi toda la arboleda de la imagen, excepto los píxeles incluidos por error en las otras clases ya comentadas, y además incluye, también por error, algunos píxeles del puente, aunque escasos. Las clases mostradas en las figuras 51 a 53 se han unido en una sola imagen mostrada en la figura 56, por conveniencia, para una visualización global de todos los píxeles que se han identificado como algún tipo de cultivo.



*Figura 56: Unión de vegetación tipos I, II y III*



Comparando estos resultados con los expuestos anteriormente obtenidos mediante el método de referencia [1], en relación a la identificación de vegetación, observamos que:

1. Se identifica la vegetación débil que el método de referencia clasificaba como un tipo de suelo. Corresponde a la representada en la figura 54. Quizá se trata de la mala hierba sobre la que hay que actuar.
2. También se clasifican como vegetación algunos píxeles del puente, aunque en menor medida en cuanto al número, y en un grupo que no es de interés para los objetivos del proyecto, junto a la arboleda.

Además, respecto a la diferenciación de varios tipos de vegetación, parece que, además de distinguir más grupos diferentes de vegetación, la clasificación obtenida por el método propuesto en la imagen de referencia responde más a tipos reales de vegetación, o al menos a distintos estados de crecimiento de la misma planta:

1. La diferenciación de la arboleda es más precisa, está prácticamente aislada en una clase aparte, excepto algunos píxeles que se escapan, pero en cantidad considerablemente menor que en el método de referencia.
2. Los pastos más frondosos en la colina a la derecha de la imagen siguen mezclándose con algunos cultivos más bajos, y con las hierbas más crecidas (figura 53), aunque en el método propuesto se distinguen de los cultivos más crecidos, que están clasificados en otros grupos (figuras 51 y 52).

En definitiva, para la imagen de referencia utilizada, parece que los resultados del método propuesto en el presente trabajo son algo más precisos que los obtenidos mediante el método de referencia, tanto en cuanto a la capacidad de identificar estructuras vegetales, donde por ejemplo se ha reducido el número de píxeles del puente clasificados como vegetación, y no se ha descartado vegetación como un tipo de suelo; como en cuanto a la precisión a la hora de distinguir los diferentes tipos, por ejemplo en el caso de la arboleda.

## 5. Conclusiones

Del trabajo realizado puede llegarse a varias conclusiones respecto a los diferentes aspectos tratados en relación a la segmentación de texturas.

En primer lugar, y por seguir el orden en el que se han mencionado en la presente memoria, respecto a la elección del espacio de colores se confirma lo afirmado en [6] sobre la criticidad de una correcta elección del espacio de color adecuado. Se ha comprobado en los experimentos realizados que los resultados de clasificación son más intuitivos trabajando con un espacio de colores perceptualmente uniforme, aunque dicho artículo se limitaba al espacio HSI. En nuestro caso, trabajando con CIELab conseguimos que:

- La distancia matemática entre dos colores diferentes se corresponda más fielmente con la diferencia percibida visualmente por un humano al comparar los mismos (véase ejemplo en sección 3.1.2)
- La luminosidad no sea necesaria para obtener buenos resultados de clasificación

Ambos puntos influyen en que la clasificación sea más acorde a la esperada según la percepción visual de la imagen, que corresponde a los elementos naturales reales presentes en la misma. Además el segundo punto nos protege en cierta medida de condiciones ambientales adversas habituales en imágenes naturales; e incluso disminuye la potencia de procesamiento necesaria respecto a RGB al trabajar sólo con dos canales.

En segundo lugar, respecto a la posibilidad de realizar un análisis espectral sobre imágenes naturales, puede concluirse, por los resultados obtenidos, que es perfectamente posible hacerlo obteniendo buenos resultados de clasificación. Se abre aquí una puerta a futuras investigaciones, incluso al aprendizaje supervisado. No obstante, también se ha observado la criticidad del diseño del banco de filtros empleado. El implementado en el presente proyecto es el que mejor resultado ha ofrecido para el conjunto de imágenes analizadas tras una gran cantidad de pruebas con diferentes variaciones de parámetros, pero probablemente un estudio más detallado y cuidadoso matemáticamente hablando del problema permitiría encontrar un diseño más óptimo que ofreciera mejores resultados.

Además, en cuanto a la eficiencia, hemos comprobado que aplicar un filtro de Gabor puede reducirse a la convolución de filtros lineales precalculados.

Por último, respecto al agrupamiento en clases de las texturas descritas, vemos cierta problemática con la aplicación del K-Means, sobre todo respecto a dejar en manos del usuario decidir cuántos grupos hay que identificar. Por un lado, formar demasiadas clases introduce una complejidad innecesaria, pero por otro, considerar un número menor implica que los resultados no son intuitivos

debido a que se "fuerza" al algoritmo a incluir píxeles que no deberían pertenecer a ninguno de los grupos en alguno de ellos.



Una solución a este respecto podría ser la propuesta en la sección 4.3 y en [18] de ir iterando sobre diferentes valores de  $K$  hasta conseguir el valor óptimo. Pero quizá aún más adecuado sería probar otros algoritmos de agrupamiento más sofisticados que sean capaces de ajustar ellos mismos el número de clases a la complejidad de la imagen.

## 6. Trabajo futuro



La principal fuente de imprecisiones en el método presentado proviene del empleo del algoritmo K-Means para el agrupamiento. Si bien es verdad que es un algoritmo simple y rápido, de gran eficiencia, quizá pudiera mejorarse la eficacia de la clasificación utilizando algún método más sofisticado capaz de ajustar el número de grupos necesarios a la complejidad de la imagen analizada.

Además, sería muy interesante comprobar el efecto de utilizar medidas de distancia alternativas a la hora de agrupar los vectores de características. De especial interés sería probar con la EMD [7][10][11][12] o con la distancia de Mahalanobis [17][18], y analizar qué efectos provocan.

En cuanto a la caracterización en sí de las texturas, el diseño del banco de filtros utilizado en este trabajo ha sido resultado meramente de un proceso iterativo de prueba y error. Es de esperar que un estudio cuidadoso de los parámetros más adecuados al tipo de imágenes tratados seguramente llevaría a ~~un aumento definitivo~~ en la precisión de la clasificación. En cuanto a la variabilidad de las texturas en las imágenes naturales, existe aún una serie de problemas a atacar a la hora de analizarlas por regiones, como son:

1. La escala básica de algunas texturas puede ser y será con frecuencia mayor que la mayor escala utilizada en los filtros de Gabor, lo que impide que los mismos sean capaces de capturar correctamente la misma.
2. Siempre habrá píxeles frontera en los cuales el filtro aplicado abarcará zonas con texturas diferentes, lo que producirá información no válida ~~y que incluso puede~~ introducir ruido en el espacio de características.
3. Condiciones de ruido debidas a iluminación, desenfoques y similares aumentan aún más la variabilidad en la imagen.
4. Incluso en condiciones ideales, las texturas naturales se caracterizan por una gran variabilidad espacial, siendo regulares sólo desde un punto de vista estadístico.

Estos puntos son los que se plantean como grandes retos y objetivos de trabajo futuro. Existen ya ciertas propuestas, como por ejemplo el pretratamiento de las imágenes expuesto en [7] y [11] mediante un método de suavizado preservando los bordes en el espacio de características. Sería interesante comprobar los resultados de su aplicación en nuestro problema.



## 7. Listado de acrónimos

- CIE: Commission Internationale de l'Eclairage
- CMYK: Cyan, Magenta, Yellow, black
- CRT: Cathode Ray Tube
- EMD: Earth Mover's Distance
- HSI: Hue, Saturation, Intensity
- HSL: Hue, Saturation, Lightness
- RGB: Red, Green, Blue

## 8. Referencias

- [1] M. Guijarro, G. Pajares , I. Riomoros , P.J. Herrera , X.P. Burgos-Artizzu , A. Ribeiro; **Automatic segmentation of relevant textures in agricultural images**; Computers and Electronics in Agriculture 75 (2011) 75–83
- [2] Robot Fleets for Highly Effective Agriculture and Forestry Management; <http://www.rhea-project.eu/>
- [3] G. Pajares, M. Guijarro, P.J. Herrera, A. Ribeiro; **Combining classifiers through fuzzy cognitive maps in natural images**;
- [4] María Guijarro, Raquel Abreu, Gonzalo Pajares; **On combining Learning Vector Quantization and the Bayesian classifiers for natural textured images**; II Congreso Español de Informática, IV Taller de Minería de Datos y Aprendizaje, 196-201
- [5] Naotoshi Seo; **Texture Segmentation using Gabor Filters**; ENEE731 Project, 2006
- [6] Leena Lepistö, Iivari Kunttu, Jorma Autio, Ari Visa; **Classification Method for Colored Natural Textures Using Gabor Filtering**;
- [7] Yossi Rubner; **Perceptual metrics for database image navigation**; PhD Thesis 1999
- [8] Thomas P. Weldon, William E. Higgins, and Dennis F. Dunn; **Efficient Gabor filter design for texture segmentation**;
- [9] Yong Huang, Kap Luk Chan, Zhihua Zhang; **Texture classification by multi-model feature integration using Bayesian networks**;
- [10] Yossi Rubner, Jan Puzicha, Carlo Tomasi, Joachim M. Buhmann; **Empirical Evaluation of Dissimilarity Measures for Color and Texture**;
- [11] Yossi Rubner, Carlo Tomasi; **Coalescing Texture Descriptors**; ARPA Image Understanding Workshop 1996
- [12] Yossi Rubner, Carlo Tomasi; **Texture Metrics**;
- [13] V. Shiv Naga Prasad, Justin Domke; **Gabor Filter Visualization**;
- [14] D. Puig, S. Alvarez, P. Sobrevilla, E. Montseny; **Analysis of natural textures using fuzzy techniques**;
- [15] Ramesh Jain, Rangachar Kasturi, Brian G. Schunck ; **Machine Vision**; McGraw Hill 1995 (Capítulo 7, Texture)
- [16] Javier R. Movellan; **Tutorial on Gabor Filters**;
- [17] Gonzalo Pajares y Jesús M. de la Cruz; **Visión por computador: imágenes digitales y aplicaciones**; 2ª edición, RA-MA, Madrid, 2007
- [18] Gary Bradski and Adrian Kaehler; **Learning OpenCV**; O'Reilly, September 2008, Primera Edición.

## **Anexo A: Código fuente**

La implementación del método expuesto utilizada en las pruebas fue escrita en lenguaje C, haciendo uso intensivo de la librería OpenCV.

No se incluye aquí el código fuente por no considerarlo necesario, pero el mismo está alojado en un repositorio privado en Bitbucket ([www.bitbucket.com](http://www.bitbucket.com)) y disponible bajo petición a [mafraba@gmail.com](mailto:mafraba@gmail.com).