



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería en Informática



TFG del Grado en Ingeniería Informática

**Sistema de reconocimiento
automático en arqueobotánica**



Presentado por Jaime Sagüillo Revilla
en Universidad de Burgos — 14 de marzo de 2017
Tutores: Álgvar Arnaiz González, José Francisco Díez
Pastor y Virginia Ahedo García



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería en Informática



Álvar Arnaiz González, José Francisco Díez Pastor profesores del departamento de Departamento de Ingeniería Civil, Área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Jaime Sagüillo Revilla, con DNI 12782524-K, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado Sistema de reconocimiento automático en arqueobotánica.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 14 de marzo de 2017

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor

Resumen

Descriptores

Arqueobotánica, Fitolito, Reconocimiento de imagenes, Python

Abstract

Keywords

Índice general

Índice general	III
Índice de figuras	IV
Índice de tablas	V
Introducción	1
Objetivos del proyecto	2
Conceptos teóricos	3
3.1. Inteligencia artificial	3
3.2. Reconocimiento de patrones	3
3.3. Aprendizaje supervisado y no supervisado	3
3.4. Segmentación	3
3.5. Binarización	4
3.6. Thresholding	4
3.7. Descriptores visuales	4
3.8. Máquinas de vectores soporte	5
Técnicas y herramientas	6
Aspectos relevantes del desarrollo del proyecto	7
5.1. Procesamiento de imágenes	7
5.2. Clasificadores	8
Trabajos relacionados	13
Conclusiones y Líneas de trabajo futuras	14
Bibliografía	15

Índice de figuras

5.1. Conversión de la imagen original a escala de grises	8
5.2. Distintos ejemplos de una imagen segmentada	9
5.3. Resultados obtenidos mediante transformación divisoria	10
5.4. Resultados tras aplicar el clasificador sobre una imagen	12

Índice de tablas

Introducción

Objetivos del proyecto

Conceptos teóricos

Para la comprensión de este proyecto será necesario la comprensión de algunos conceptos teóricos que introduciré en este apartado:

- Inteligencia artificial
- Reconocimiento de patrones
- Aprendizaje supervisado y no supervisado
- Segmentación
- Binarización
- *thresholding*
- Descriptores visuales
- Maquinas de vector soporte

3.1. Inteligencia artificial

3.2. Reconocimiento de patrones

3.3. Aprendizaje supervisado y no supervisado

3.4. Segmentación

La segmentación en el campo de la visión artificial, como se indica en la wikipedia, consiste en subdividir una imagen en varios pixeles u objetos. [4] Cuando segmentamos una imagen, lo que pretendemos hacer es cambiar su representación para poder obtener de esta una mayor utilidad o cantidad de información.

En nuestro caso, segmentamos la imagen para eliminar el fondo de ella y obtener así una imagen con solo su parte delantera. De esta manera, eliminamos el ruido que existe en la imagen y, a su vez, la simplificamos reteniendo la parte de la imagen en la que se encuentran los objetos que nos interesan.

Posteriormente a ese paso, nos interesa, como es obvio, dividir la parte delantera de la imagen resultante en objetos. De este modo, obtendremos cada uno de los objetos por separado de forma idónea.

3.5. Binarización

La binarización de una imagen consiste en la simplificación de los valores de cada pixel a 2 posibles valores, blanco o negro, representando el fondo y el frente de la imagen cada uno de ellos. Esta técnica nos permite conservar únicamente la información que nos interesa, eliminando el resto.

3.6. Thresholding

Es el método mas simple para la segmentación de una imagen, pudiendose utilizar para la binarización de una imagen, como es nuestro caso. Consiste en reemplazar los píxeles por debajo de una determinada constante a píxeles negros, y los que se encuentran por encima a píxeles blancos o viceversa.

Existen distintas maneras de llevar a cabo este proceso, siendo uno de lo más conocidos el método de Otsu. [5]

3.7. Descriptores visuales

Los descriptores visuales, o descriptores de características, son descripciones de las características visuales de los contenidos en imágenes o videos, en nuestro caso de imágenes, con el proposito de la detección de objetos [2]. El objetivo de los descriptores visuales es obtener la información que resulta significativa, eliminando a su vez la que no lo es. Así, utilizaremos la información que el descriptor nos proporciona para detectar los objetos que nos interesan en una imagen. Algunos ejemplos de características son la forma, el color o la textura.

Como se puede imaginar, obtener las características a mano es una tarea complicada y que usualmente no funciona correctamente. Por ello, utilizamos un método de extracción automática de características como es *Histogram of Oriented Gradients*, el cual se basa en los gradientes de la imagen para detectar los distintos objetos que se encuentran en la imagen [3].

3.8. Máquinas de vectores soporte

Las máquinas de vectores soporte, o SVM, son modelos de aprendizaje supervisados utilizados para tareas de clasificación o de regresión [1]. En nuestro caso este modelo se ve usado para tareas de clasificación, puesto que es lo que nos concierne en nuestra problemática.

Para que nuestra SVM sea capaz de clasificar los objetos, le proveemos de un conjunto de entrenamiento compuesto por positivos y negativos, es decir, ejemplos de los objetos que nos interesan y otros objetos, respectivamente. A partir de esta información nuestra SVM será capaz de clasificar nuevos ejemplos en las categorías pertinentes.

Técnicas y herramientas

Aspectos relevantes del desarrollo del proyecto

5.1. Procesamiento de imágenes

Como primera aproximación al problema que nos concierne, nos hemos enfrentado al procesamiento de imágenes mediante la librería Scikit-image para Python. Mediante esta herramienta trataremos de dar solución a nuestro problema siguiendo los siguientes pasos:

1. Convertimos la imagen a escala de grises
2. Segmentamos los objetos del fondo de la imagen
3. Obtenemos los distintos objetos de la imagen

Convertimos la imagen a escala de grises

La conversión de la imagen original (RGB) a escala de grises viene motivada con el mero objetivo de poder segmentar los objetos del fondo de la imagen mediante el método de *Thresholding*. Solo pudiéndose partir de una imagen en escala de grises. En la figura 5.1 podemos ver los resultados.

Segmentamos los objetos del fondo de la imagen

Una vez tenemos la imagen en escala de grises, procedemos a transformar nuestra imagen en una imagen en blanco y negro o binarizada. Los motivos por los que binarizamos la imagen es para obtener una imagen que sea más significativa para nosotros y además este simplificada. Lo cual nos sera útil para facilitarnos su procesamiento.

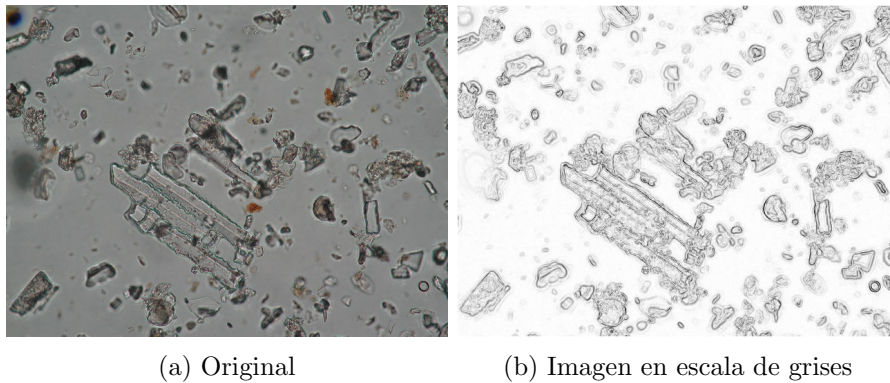


Figura 5.1: Conversión de la imagen original a escala de grises

Scikit nos proporciona distintos métodos mediante los cuales podemos segmentar una imagen. En la figura 5.2 podemos observar el resultado aplicando distintos métodos, los cuales se van indicando en cada una de las figuras.

Obtenemos los distintos objetos de la imagen

Después de tener la imagen binarizada de la forma más apropiada posible probamos a segmentar los distintos objetos de nuestra imagen.

Transformación divisoria

Transformación divisoria, o en inglés *Watershed segmentation*, es un algoritmo clásico para la segmentación de objetos en una imagen.

Durante las primeras pruebas, la segmentación más interesante hasta el momento ha sido la que se muestra en la figura 5.3. A partir de *Watershed segmentation* con marcado. Más allá de esta segmentación no se ha conseguido nada mejor.

5.2. Clasificadores

La aproximación mediante procesamiento de imágenes no parece la más adecuada visto los resultados obtenidos. Por ello vamos a realizar el estudio sobre una segunda aproximación mediante clasificadores, junto a descriptores visuales.

El primer conjunto de técnicas escogido ha sido una Máquina de Vector Soporte junto al Histograma de los gradientes orientados, la cual es una técnica para la extracción automática de características. Sin embargo, iremos utilizando distintas técnicas para adoptar la que mejor se adapte a nuestra problemática.

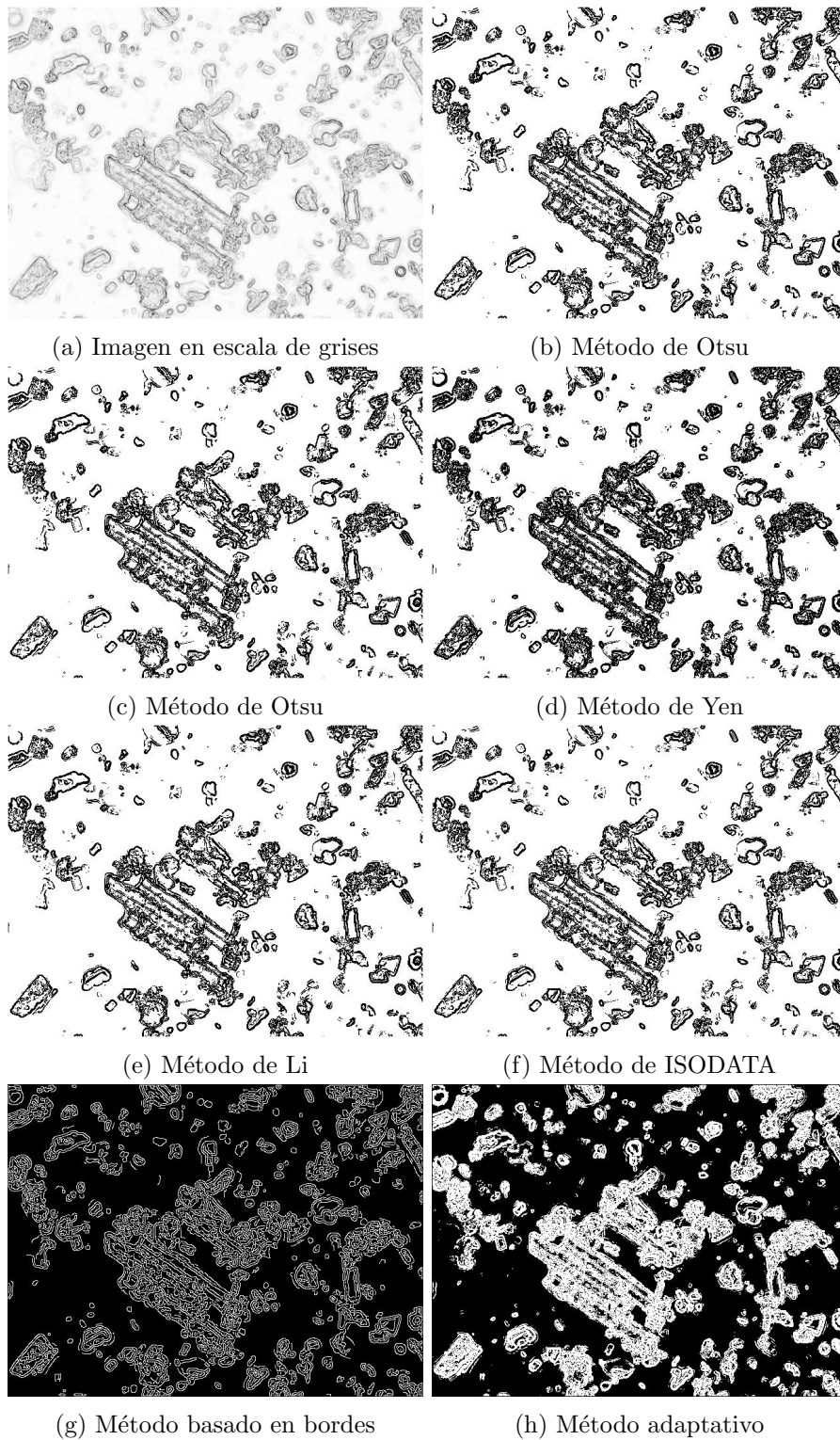


Figura 5.2: Distintos ejemplos de una imagen segmentada

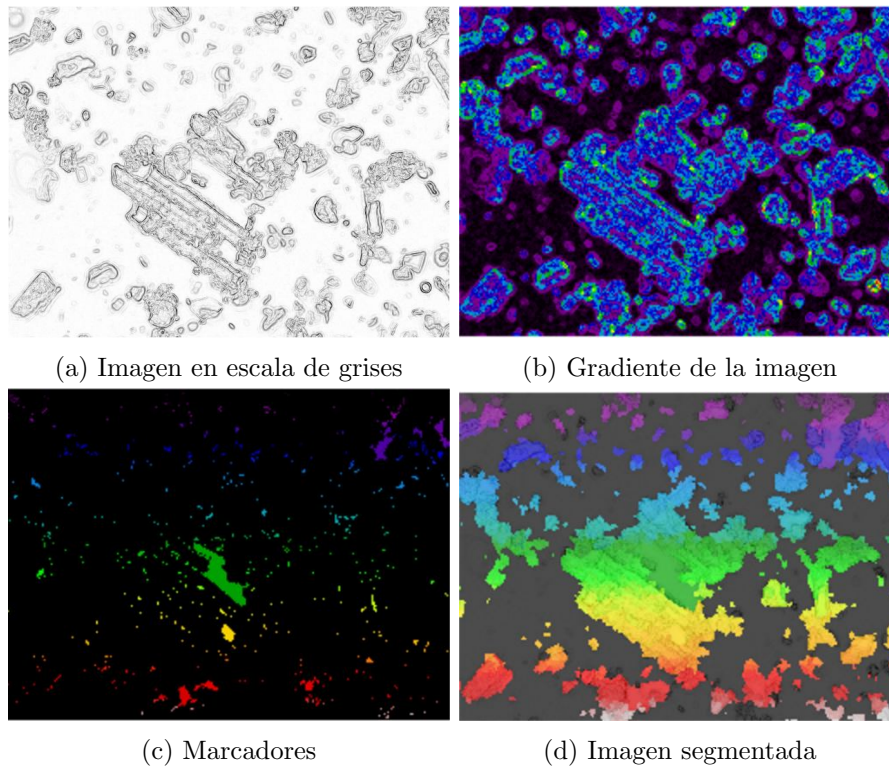


Figura 5.3: Resultados obtenidos mediante transformación divisoria

El procedimiento para obtener el clasificador es el siguiente:

1. Crear un conjunto de entrenamiento de imágenes de caras que consideramos que son elementos positivos.
2. Crear un conjunto de entrenamiento de imágenes de no-caras que consideramos que son negativos.
3. Extraer las características del conjunto de entrenamiento mediante un descriptor visual.
4. Entrenar el clasificador.

Finalizado el entrenamiento, ya tenemos nuestro clasificador listo para enviarle nuevas imágenes y que sean clasificadas.

Reconocimiento de imágenes en nuevas imágenes

Para el reconocimiento de objetos en nuevas imágenes, deberemos llevar a cabo los tres siguientes pasos:

1. Dividir la imagen en múltiples fragmentos.
2. Comprobar si cada uno de los fragmentos contiene el objeto.
3. Si existe solapamiento en la detección de objetos, muy común en el uso de este tipo de clasificadores, se deben de combinar dichos solapamientos en uno único.

Cada uno de los fragmentos anteriores se solapa en gran medida. Por lo que origina un problema de sobreconocimiento de objetos, reconociendo donde existe un posible positivo, más de uno, en la mayoría de casos. Por ello, se aplica el tercer paso sobre los objetos reconocidos, que es la eliminación del solapamiento de objetos mediante la técnica de *Non-Maximum supresion*.

Aplicación sobre el reconocimiento de caras

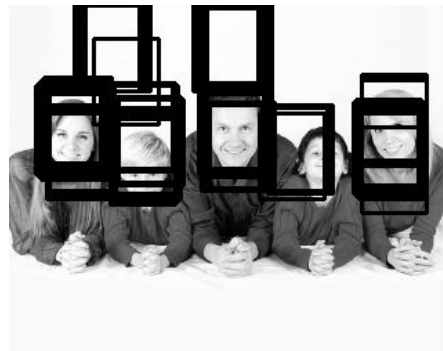
Como previa experimentación con esta metodología, vamos a entrenar el clasificador para el reconocimiento de caras. Y en función de la efectividad del método sobre las caras tomaremos una serie de conclusiones sobre las que decidiremos si llevar a cabo esta solución sobre nuestro problema.

Como explicábamos anteriormente, una vez tenemos nuestro clasificador le enviamos una nueva imagen, como podría ser la presentada en la figura 5.4a. A partir de esta imagen el clasificador nos permitirá obtener las ventanas ¹ en las que detecta una cara, como vemos en la figura 5.4b. Podemos apreciar que existe más de una ventana alrededor de cada cara. Y finalmente, tras aplicar el método *Non-Maximum supresion* obtenemos el resultado final mostrado en la figura 5.4c.

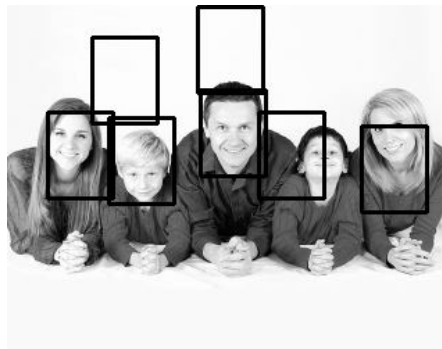
¹Se entiende por ventana , en este contexto, a la caja o cuadrilatero que etiqueta un positivo en una imagen.



(a) Imagen original



(b) Imagen después de aplicarla nuestro clasificador



(c) Imagen tras aplicarla *Non-Maximum supresion*

Figura 5.4: Resultados tras aplicar el clasificador sobre una imagen

Trabajos relacionados

Conclusiones y Líneas de trabajo futuras

Bibliografía

- [1] Wikipedia. Support vector machine — Wikipedia, the free encyclopedia, 2016. [Online; accedido 22-Febrero-2017].
- [2] Wikipedia. Visual descriptor — Wikipedia, the free encyclopedia, 2016. [Online; accedido 22-Febrero-2017].
- [3] Wikipedia. Histogram of oriented gradients — Wikipedia, the free encyclopedia, 2017. [Online; accedido 22-Febrero-2017].
- [4] Wikipedia. Image segmentation — Wikipedia, the free encyclopedia, 2017. [Online; accedido 10-Febrero-2017].
- [5] Wikipedia. Otsu's method — Wikipedia, the free encyclopedia, 2017. [Online; accedido 8-Febrero-2017].