



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería en Informática



TFG del Grado en Ingeniería Informática

Sistema de reconocimiento
automático en arqueobotánica
Documentación Técnica



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

Índice general

Índice general	I
Índice de figuras	III
Índice de tablas	IV
Apéndice A Planificación	1
A.1. Introducción	1
A.2. Estudio previo	1
A.3. Planificación temporal	3
A.4. Estudio de viabilidad	3
Apéndice B Especificación de Requisitos	4
B.1. Introducción	4
B.2. Objetivos generales	4
B.3. Catalogo de requisitos	4
B.4. Especificación de requisitos	4
Apéndice C Especificación de diseño	5
C.1. Introducción	5
C.2. Diseño de datos	5
C.3. Diseño procedimental	5
C.4. Diseño arquitectónico	5
Apéndice D Documentación técnica de programación	6
D.1. Introducción	6
D.2. Estructura de directorios	6
D.3. Manual del programador	6
D.4. Compilación, instalación y ejecución del proyecto	6
D.5. Pruebas del sistema	6

<i>ÍNDICE GENERAL</i>	II
Apéndice E Documentación de usuario	7
E.1. Introducción	7
E.2. Requisitos de usuarios	7
E.3. Instalación	7
E.4. Manual del usuario	7
Bibliografía	8

Índice de figuras

A.1. Burndown de la <i>sprint</i> 0	2
A.2. Burndown del <i>sprint</i> 1	3

Índice de tablas

Apéndice A

Planificación

A.1. Introducción

Para llevar a cabo este proyecto vamos a aplicar una metodología llamada Scrum. Scrum es una metodología de desarrollo software ágil, es decir, durante cada *sprint*¹ se asignarán unas determinadas tareas a cumplimentar, con un producto como consecuencia de estas tareas. Al final de cada *sprint* se realizará una reunión junto a los tutores para validar los avances realizados y determinar las tareas a realizar durante el siguiente *sprint*.

A.2. Estudio previo

Sprint 0

Estas son las tareas a realizar durante este *sprint* 0:

- Probar L^AT_EX
- Gestor de tareas/versiones: Github y Zenhub
- Instalar Anaconda y Jupyter
- Leer los artículos propuestos por los tutores
- Comenzar a probar algunos algoritmos de binarización

Como se puede ver las tareas a realizar son básicas, puesto que es el *sprint* 0 y es un *sprint* de mera adaptación al entorno de trabajo. La única tarea que supone un esfuerzo de comprensión mayor es la lectura de los artículos

¹*Sprint*: es el período en el cual se lleva a cabo el trabajo en sí.[1]

propuestos sobre trabajos relacionados o con una problemática similar a la nuestra. A continuación, en la figura A.1, se muestra el diagrama *burndown* de este *sprint*.



Figura A.1: Burndown de la *sprint* 0

Sprint 1

Estas son las tareas a realizar durante esta *sprint* 1:

- Documentar lo realizado durante el *sprint* 0
- Documentar lo que se irá realizando durante este *sprint* 1
- Continuar probando con algoritmos de procesamiento de imágenes
- Probar una aproximación con clasificadores al problema

Puesto que en el *sprint* anterior no se documentó lo realizado, durante este se pretende documentar todo lo realizado durante el *sprint* anterior y este. Además de continuar probando con algoritmos de procesamiento de imágenes y comenzar a probar con la aproximación al problema mediante clasificadores.

En este *sprint* me vi desbordado de trabajo debido a la subestimación del esfuerzo a empeñar en las distintas tareas. No siendo capaz de comenzar a probar una aproximación con clasificadores. Por ello la tarea «Probar una aproximación con clasificadores al problema» se vio movida al siguiente *sprint*.

A continuación, en la figura A.2, se muestra el diagrama *burndown* de este *sprint*. El cual tiene dicho aspecto debido a que muchas de las tareas se trabajaron de manera paralela, no siendo acabadas hasta el final del *sprint*. Y, además, otras de las tareas no fueron cerradas cuando se debió, aspecto que se corregirá en los siguientes *sprints*.

Figura A.2: Burndown del *sprint* 1

Sprint 2

Estas son las tareas a realizar durante este *sprint* 2:

- Probar una aproximación con clasificadores al problema
- Aplicación del método "Non maximum suppression" sobre el clasificado

Puesto que la aproximación mediante reconocimiento de imágenes no reflejaba unos resultados muy positivos, durante la reunión mantenida con los tutores se decidió el uso de una técnica distinta. Nos referimos a la utilización de un clasificador, junto a un descriptor visual.

Debido a que todavía no se poseían suficientes imágenes para el estudio del problema mediante esta técnica, lo que se decidió es aplicarla sobre otro problema de características similares, como es el reconocimiento de caras en imágenes. Con unos resultados bastante positivos debido a distintos razonamientos explicados en la Memoria, sección de Aspectos relevantes del proyecto.

A.3. Planificación temporal

A.4. Estudio de viabilidad

Viabilidad económica

Viabilidad legal

Apéndice B

Especificación de Requisitos

- B.1. Introducción
- B.2. Objetivos generales
- B.3. Catalogo de requisitos
- B.4. Especificación de requisitos

Especificación de diseño

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico

Documentación técnica de programación

- D.1. Introducción
- D.2. Estructura de directorios
- D.3. Manual del programador
- D.4. Compilación, instalación y ejecución del proyecto
- D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

- E.1. Introducción**
- E.2. Requisitos de usuarios**
- E.3. Instalación**
- E.4. Manual del usuario**

Bibliografía

- [1] Wikipedia. Scrum — Wikipedia, the free encyclopedia, 2017. [Online; accedido 12-Marzo-2017].