



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería  
Informática**

**Transformaciones invariantes a  
la iluminación para mejora de  
la segmentación en imágenes.  
Aplicación a piezas metálicas.**



Presentado por Jonás Martínez Sanllorente  
en Universidad de Burgos — 23 de junio  
de 2024

Tutor: Dr. Carlos López Nozal y Dr. Pedro  
Latorre Carmona



---

# Índice general

---

<b>Índice general</b>	<b>i</b>
<b>Índice de figuras</b>	<b>iii</b>
<b>Índice de tablas</b>	<b>iv</b>
<b>Apéndice A Plan de Proyecto Software</b>	<b>1</b>
A.1. Introducción . . . . .	1
A.2. Planificación temporal . . . . .	2
A.3. Estudio de viabilidad . . . . .	10
<b>Apéndice B Especificación de Requisitos</b>	<b>13</b>
B.1. Introducción . . . . .	13
B.2. Objetivos generales . . . . .	14
B.3. Catálogo de requisitos . . . . .	14
B.4. Especificación de requisitos . . . . .	17
<b>Apéndice C Especificación de diseño</b>	<b>45</b>
C.1. Introducción . . . . .	45
C.2. Diseño de datos . . . . .	45
C.3. Diseño procedimental . . . . .	45
C.4. Diseño arquitectónico . . . . .	46
<b>Apéndice D Documentación técnica de programación</b>	<b>47</b>
D.1. Introducción . . . . .	47
D.2. Estructura de directorios . . . . .	47
D.3. Manual del programador . . . . .	48

D.4. Compilación, instalación y ejecución del proyecto . . . . .	52
D.5. Pruebas del sistema . . . . .	56
<b>Apéndice E Documentación de usuario</b>	<b>61</b>
E.1. Introducción . . . . .	61
E.2. Requisitos de usuarios . . . . .	61
E.3. Instalación . . . . .	62
E.4. Manual del usuario . . . . .	63
<b>Apéndice F Anexo de sostenibilización curricular</b>	<b>71</b>
F.1. Introducción . . . . .	71
<b>Bibliografía</b>	<b>73</b>

---

# Índice de figuras

---

A.1. Burndown del sprint 4. . . . .	5
A.2. Burndown del sprint 5. . . . .	6
A.3. Burndown del sprint 6. . . . .	7
A.4. Burndown del sprint 7. . . . .	8
B.1. Diagrama de casos de uso. . . . .	18
C.1. Diagrama de secuencia de los distintos componentes de la aplicación. . . . .	46
D.1. Instalación Statistics and Machine Learning Toolbox . . . . .	50
D.2. Descarga del repositorio mediante el entorno web. . . . .	51
D.3. Ventana principal de MATLAB App Designer. . . . .	52
D.4. Icono de compilación y ejecución de MATLAB App Designer. . . . .	53
D.5. Ventana de la aplicación compilada y ejecutada. . . . .	54
D.6. Creación de MATLAB Standalone App. . . . .	55
D.7. Resultados de la ejecución de los tests. . . . .	59
E.1. Instalación de la release desde GitHub. . . . .	62
E.2. Apartado de bienvenida de InvIMP. . . . .	63
E.3. Apartado de exploración de algoritmos de InvIMP. . . . .	64
E.4. Apartado de histórico de exploraciones de InvIMP. . . . .	68
E.5. Apartado de ayuda de InvIMP. . . . .	69

---

# Índice de tablas

---

A.1. Desglose de tiempo dedicado al proyecto. . . . .	9
A.2. Costes de hardware. . . . .	10
A.3. Costes de software. . . . .	10
A.4. Costes de personal. . . . .	11
A.5. Costes totales. . . . .	11
B.1. CU-1 Gestión de transformación invariante. . . . .	19
B.2. CU-2 Seleccionar imagen. . . . .	20
B.3. CU-3 Imagen del sistema. . . . .	21
B.4. CU-4 Carga automática de la imagen ground truth. . . . .	22
B.5. CU-5 Imagen propia del usuario. . . . .	23
B.6. CU-6 Seleccionar una imagen ground truth propia. . . . .	24
B.7. CU-7 Modificar imagen ground truth seleccionada. . . . .	25
B.8. CU-8 Modificar imagen seleccionada. . . . .	26
B.9. CU-9 Seleccionar algoritmo invariante. . . . .	27
B.10.CU-10 Seleccionar algoritmo de agrupamiento. . . . .	28
B.11.CU-11 Seleccionar número de centros. . . . .	29
B.12.CU-12 Almacenar resultados en caché. . . . .	30
B.13.CU-13 Recuperar resultados almacenados en caché. . . . .	31
B.14.CU-14 Guardar imágenes resultantes. . . . .	32
B.15.CU-15 Gestión de caché. . . . .	33
B.16.CU-16 Visualizar información de antiguas ejecuciones en formato tabla. . . . .	34
B.17.CU-17 Poder reordenar las columnas. . . . .	35
B.18.CU-18 Visualizar la imagen correspondiente a la fila seleccionada. . . . .	36
B.19.CU-19 Abrir el directorio de la caché. . . . .	37
B.20.CU-20 Borrar contenido de la caché. . . . .	38
B.21.CU-21 Configuración. . . . .	39

B.22.CU-22 Seleccionar el idioma. . . . .	40
B.23.CU-23 Ayuda. . . . .	41
B.24.CU-24 Documentación dentro de la aplicación. . . . .	42
B.25.CU-25 Tooltips. . . . .	43





## *Apéndice A*

---

# Plan de Proyecto Software

---

## A.1. Introducción

La fase de planificación es esencial en cualquier proyecto. Durante esta etapa se calculan el trabajo, el tiempo y el dinero necesarios para completar el proyecto. Para ello, se analiza detalladamente cada una de las partes que lo componen lo que permite conocer los recursos necesarios para su realización. En este anexo se describe todo este proceso.

La fase de planificación se puede dividir en dos partes:

- **La planificación temporal:** se crea un calendario o cronograma. Aquí se calcula el tiempo necesario para completar cada una de las partes del proyecto. Se debe establecer una fecha de inicio fija y una fecha de finalización estimada, considerando la importancia de cada tarea y los requisitos necesarios para comenzar a trabajar en ellas.
- **El estudio de viabilidad:** se enfoca en la factibilidad del proyecto y se divide en dos apartados.
  - **La viabilidad económica:** en esta parte se estiman los costos y los beneficios que podría traer la realización del proyecto.
  - **La viabilidad legal:** aquí es donde se tiene en cuenta el contexto del proyecto y toca aquella ley que le afecta. Se deben analizar todas aquellas que afecten al proyecto.

## A.2. Planificación temporal

En líneas generales, se ha seguido la metodología Scrum, aunque de manera adaptada a mi situación como único desarrollador, he mantenido ciertos principios clave.

- Se aplicó una estrategia de desarrollo incremental mediante iteraciones (sprints) y revisiones.
- La duración media de los sprints ha sido de dos semanas.
- Aunque no he tenido reuniones diarias, se han realizado revisiones periódicas al final de cada sprint para evaluar el progreso y planificar el siguiente.
- Esto ha permitido mantener la esencia de la mejora continua y la entrega incremental de valor.

He de mencionar que no es hasta la reunión de control del 26 de marzo, que empiezo a reflejar el progreso en GitHub añadiendo todas las issues de reuniones anteriores indicando S1, S2 y S3 en el propio nombre de la issue para indicar de qué semana habían sido ya que en GitHub aparecen como que todas fueron añadidas entre el 26 y el 28 de marzo. Zube lo comienzo a utilizar para organizar mejor los sprints desde el sprint 4 debido a que Carlos me recomienda utilizar dicha herramienta para tener un mejor control de estos.

### **Sprint 0 (04/03/2024 - 11/03/2024)**

La reunión que dio inicio a este sprint marcó el comienzo del proyecto de fin de carrera. Inicialmente, Pedro Latorre me había explicado en qué consistía la idea inicial del proyecto en una reunión previa. Esta idea la profundizamos un poco más, además de indicarme unas pautas a seguir a la hora de buscar el estado del arte en esta área y plantear unos objetivos a cumplir durante este primer sprint.

Los objetivos de la reunión del primer sprint fueron: crear un repositorio en GitHub para el proyecto, probar dos métodos de procesamiento de imágenes, específicamente el Análisis de Componentes Principales (PCA) y la Teoría Retinex, y guardar la imagen resultante permitiendo acceder al valor de cada píxel. También se acordó investigar y aplicar métodos de

agrupamiento de datos, como K-Means, en la imagen original y en la imagen procesada para comparar los resultados.

Además, se decidió eliminar el cuarto canal en las imágenes que lo tengan y evaluar críticamente los resultados obtenidos. Se enfatizó que no se esperen resultados inmediatos, sino que se debe evaluar y descartar posibles caminos de manera progresiva.

En este momento del proyecto simplemente creaba una serie de tareas a completar antes de la próxima semana por lo que no tengo como tal un registro temporal en Zube de cuando se completo cada tarea. Mas tarde estas tareas las pase el 26/03/2024 a GitHub por recomendación de mi tutor ya que así podía reflejar el trabajo realizado hasta el momento. Estas issues tienen en el nombre [S1].

### **Sprint 1 (11/03/2024 - 25/03/2024)**

Este sprint tuvo como objetivo: mostrar la representación del histograma de los valores de los píxeles de cada canal en la imagen resultante, determinar los valores mínimos y máximos de los píxeles, y visualizar la matriz resultante utilizando imshow. Se discutió que, aunque imshow(M, []) normaliza la imagen para visualización, imwrite guarda la imagen sin normalizar. Se acordó guardar la imagen utilizando el método saveas para conservar el rango dinámico y almacenar la matriz en un archivo .mat con el resultado de la transformación. También se destacó la importancia de guardar una imagen que se pueda visualizar fácilmente y familiarizarse con los resultados de cada método aplicado.

Se decidió añadir los métodos de agrupamiento FUZZY C-Means y GMM, y probar otros métodos de agrupamiento que tengan en cuenta el contexto y no solo el valor del píxel. Además, se acordó buscar información sobre Geoffrey Machlachlan para profundizar en el tema.

Al igual que en el sprint 0, simplemente creaba una serie de tareas a completar antes de la próxima semana por lo que no tengo como tal un registro temporal en Zube y en GitHub fueron añadidas el 26/03/2024. Estas issues tienen en el nombre [S2].

### **Sprint 2 (25/03/2024 - 04/04/2024)**

Este sprint tuvo como objetivo: pasar los algoritmos de agrupamiento de python a MATLAB ya que los tiempos de ejecución en Python el cual era el lenguaje que estaba utilizando hasta ese momento eran muy elevados. Se

decidió crear una comparación en la misma ventana dependiendo del número de centros para así poder observar las diferencias de una manera más cómoda y rápida. Además, se acordó agregar 4-5 métodos más de transformación invariante.

También se decidió generar la imagen ground truth y evaluar el resultado mediante una máscara binaria, calculando la tasa de acierto con la fórmula  $(n^{\circ} \text{ de píxeles que coinciden} / n^{\circ} \text{ total de píxeles}) * 100$ , y mostrar en la ventana el valor de esta tasa.

Al igual que en el sprint 0 y sprint 1, simplemente creaba una serie de tareas a completar antes de la próxima semana por lo que no tengo como tal un registro temporal en Zube y en GitHub fueron añadidas el 26/03/2024. Estas issues tienen en el nombre [S3].

### **Sprint 3 (04/04/2024 - 18/04/2024)**

En la reunión de preparación de este sprint es cuando Carlos López se unió como cotutor al proyecto al pedírselo Pedro Latorre, este en un futuro pasaría a ser el tutor del proyecto de fin de grado. Este sprint tuvo como objetivo: familiarizarse con Overleaf y comenzar a rellenar los tres primeros apartados del documento. Además, se acordó crear una cuenta en Zube para organizar las tareas en milestones y reorganizar los directorios del proyecto. La nueva estructura incluirá un directorio "doc" con la plantilla de Ubuvirtual y un directorio para el programa MATLAB, el cual contendrá todos aquellos métodos que venía utilizando separados en distintos scripts para que de esta manera se mejora la modularidad, reutilización y mantenibilidad del código.

Otro objetivo fue la integración de MATLAB con una aplicación web, iniciando un pequeño proyecto en GitHub para gestionar esta integración. Otro objetivo fue el desarrollo de una pequeña aplicación de escritorio de MATLAB la cual servirá para probar distintos algoritmos sobre diferentes imágenes. De esta manera además de demostrar numéricamente que hay una mejoría notable tras la aplicación de métodos invariantes sobre piezas metálicas, permitirá de una forma sencilla y visual el probar distintos métodos sobre imágenes propias.

En el anterior sprint no conseguí encontrar un método de agrupamiento con contexto espacial que suponga una mejora notable en los resultados por lo que esta issue no se cerrará y formara parte también de este cuarto sprint.

Al ser este el primer sprint en el que incorpore Zube para gestionar las issues abiertas, aun no terminaba de manejarme bien con la herramienta

por lo que aunque me ayudo con la gestión, no cree el sprint en el por lo que no tengo gráfico burndown de este sprint.

### Sprint 4 (18/04/2024 - 02/05/2024)

Es en este sprint cuando al fin comienzo a reflejar como tal los distintos sprints en Zube. Gracias a esto, a partir de este momento me ha sido más cómodo trabajar con las issues de cada sprint. Este sprint tuvo como objetivo: mejorar las funcionalidades de la aplicación y realizar cambios menores en el texto mostrado al usuario dentro de la aplicación para evitar confusiones y corregir incoherencias. Se acordó cambiar el orden en el que se muestran los resultados de la aplicación. También se discutió la importancia de eliminar las imágenes mostradas de ejecuciones anteriores.

Añadir un selector para guardar los archivos y su correspondiente mensaje de confirmación indicando que los archivos se han guardado correctamente. Además, de buscar una solución para el problema relacionado con el sistema de directorios donde se guardan los archivos.

Finalmente, se planteó la creación de otra aplicación para poder probar con imágenes propias. Al tener una imagen ground truth, se podría obtener un porcentaje de acierto de la ejecución, lo que ayudaría a evaluar mejor los resultados obtenidos.

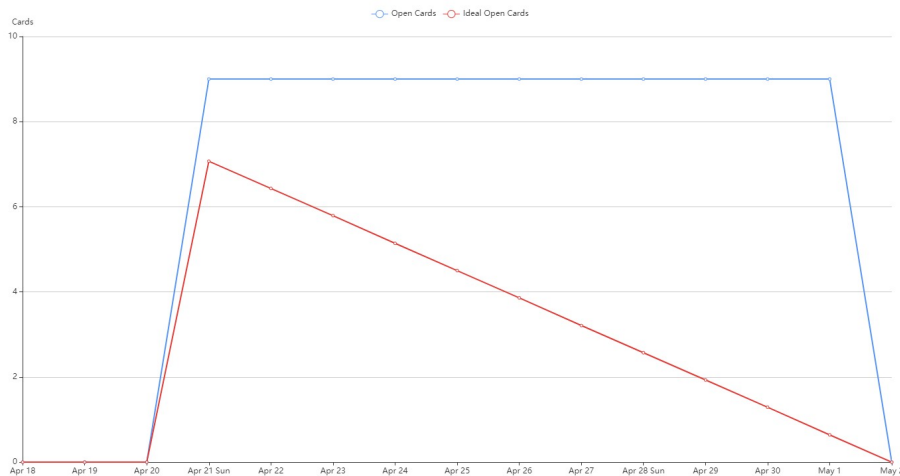


Figura A.1: Burndown del sprint 4.

Como se puede ver en el gráfico burndown [A.1](#), las issues de este sprint no es hasta el ultimo día que las marco como completadas ya que en ese

momento no era consciente que las tenía que cerrar en vez de dejarlas en revisión y ya cerrarlas tras la reunión.

## Sprint 5 (02/05/2024 - 23/05/2024)

Este sprint tuvo como objetivo: mejorar la experiencia del usuario y optimizar la funcionalidad de la aplicación. Implementar notificaciones periódicas sobre el estado de la ejecución, indicando que podría tomar mucho tiempo además de añadir la funcionalidad de borrar resultados anteriores. También se plantearon pequeños cambios en la interfaz, como eliminar el "de los porcentajes y añadir un botón para guardar imágenes.

Distintas modificaciones de la estructura de la interfaz, añadiendo menús de ayuda, una lista de nombres de imágenes en caché, un menú de bienvenida y ayuda contextual. Una actualización de los nombres de los archivos de resultados, añadiendo la fecha de ejecución al inicio. Además, de corregir errores en la selección de imágenes desde rutas diferentes a /data y una mejora de la gestión de errores.

Se llegó a la conclusión de agregar una caché de imágenes para almacenar temporalmente los resultados, cargando ejecuciones anteriores con los mismos parámetros para reducir el coste computacional. Al cerrar la aplicación, la caché se borraría. Finalmente, se planteó la corrección de algunos errores detectados en los algoritmos, una mejora de la documentación y el README del proyecto.

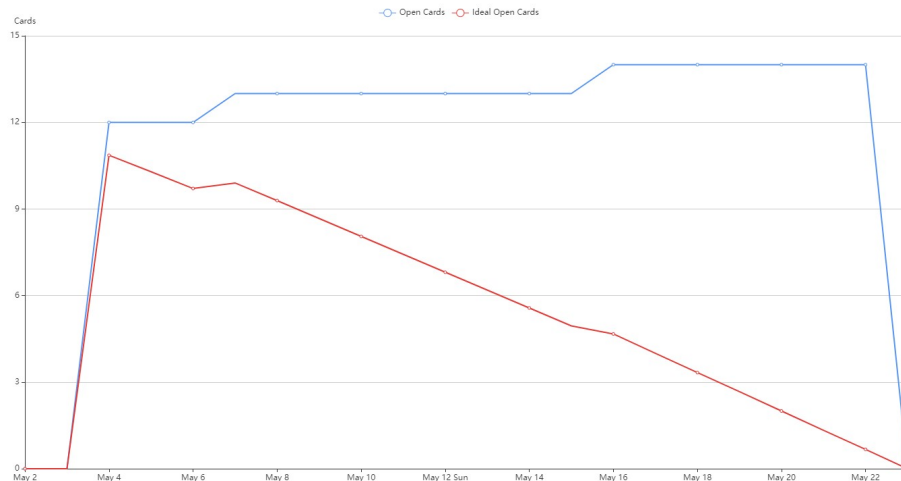


Figura A.2: Burndown del sprint 5.

Como se puede ver en el gráfico burndown A.2, las issues de este sprint no es hasta el ultimo día que las marco como completadas ya que en ese momento no era consciente que las tenia que cerrar en vez de dejarlas en revisión y ya cerrarlas tras la reunión.

### Sprint 6 (23/05/2024 - 30/05/2024)

Este sprint tuvo como objetivo: mejorar la funcionalidad y la usabilidad de la aplicación. Se planea realizar diversas mejoras en la documentación en HTML y crear un README en la carpeta de imágenes. Además, se modificará la visualización de los nombres de las imágenes dentro de la aplicación y se añadirán tooltips para una mejor comprensión del usuario.

Se establecerán condiciones numéricas en los campos de número de centros y se añadirá un límite de iteraciones máximo en los métodos de agrupamiento. También se trabajará en la caché para hacerla persistente, permitiendo borrar el historial y mejorando su escalabilidad.

En cuanto a la evaluación de la calidad, se incorporará una medida de comparación utilizando la imagen ground truth y se mejorará la nomenclatura de los algoritmos invariantes, reemplazando los nombres vacíos con “no aplicado”. Se abordarán problemas técnicos como la minimización de la app al cargar una imagen, la eliminación de la imagen ground truth al introducir una nueva que no tenga una asociada, y se revisarán soluciones para evitar que la aplicación se detenga por errores. Además, se tratará de introducir concurrencia para manejar tareas como el cancelar la ejecución.

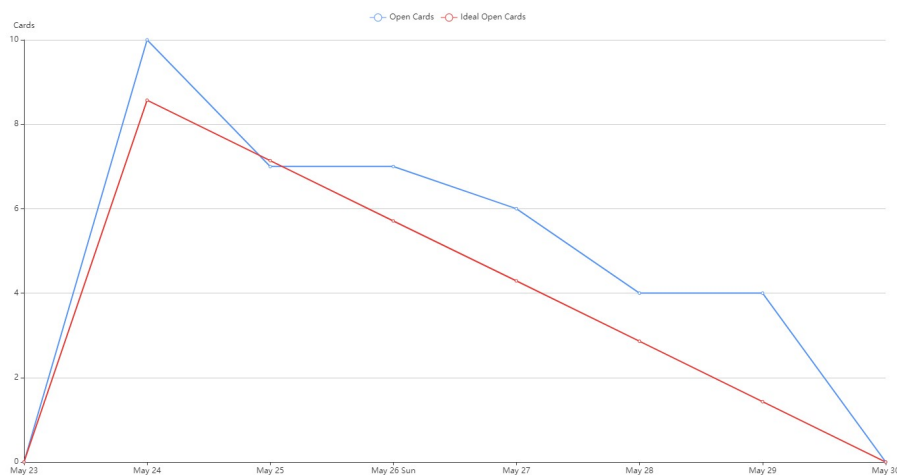


Figura A.3: Burndown del sprint 6.

Como se puede ver en el gráfico burndown [A.3](#), las issues de este sprint si que se fueron cerrando correctamente a medida que se fueron completando.

### Sprint 7 (31/05/2024 - 10/06/2024)

Este sprint tuvo como objetivo: preparar la primera release oficial de la aplicación. Se realizarán mejoras en la gestión de errores para asegurar una experiencia de usuario más robusta. Además, se tratará aplicar concurrencia para permitir la cancelación de la ejecución, la cual es una tarea que no se logró completar en el sprint anterior.

Se deshabilitarán los botones durante la ejecución de procesos para evitar acciones no deseadas y se revisará el código, actualizando la documentación interna para mantener la coherencia. También se implementará soporte bilingüe, añadiendo opciones en español e inglés para hacer la aplicación accesible a un público más amplio.

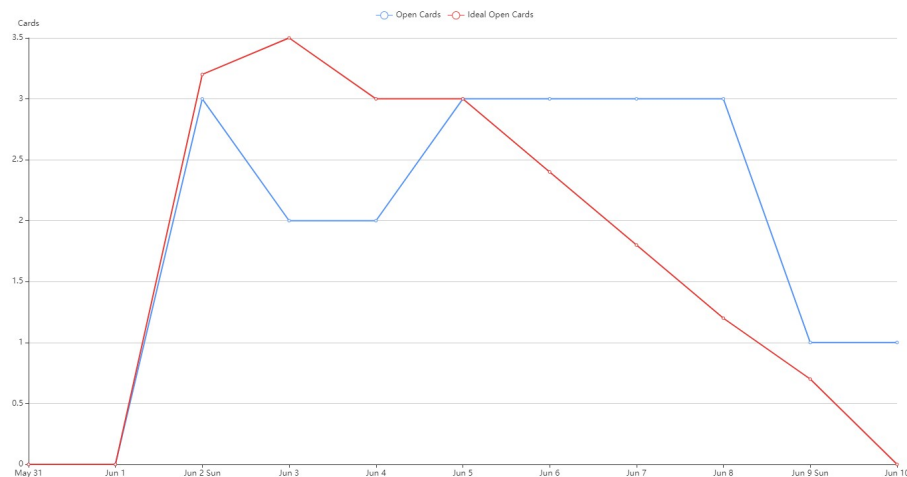


Figura A.4: Burndown del sprint 7.

Como se puede ver en el gráfico burndown [A.4](#), las issues de este sprint si que se fueron cerrando correctamente a medida que se fueron completando a excepción de una que por falta de tiempo quedo a completar en el siguiente sprint.

### Sprint 8 (11/06/2024 - 27/06/2024)

Este sprint tuvo como objetivo: implementar varias mejoras y preparar la aplicación para una release oficial. Se mejorarán los tooltips, especialmente



el de centros, explicando que es una agrupación de colores. Además, se añadirá la posibilidad de imprimir y visualizar la imagen seleccionada en la tabla de la caché, y se modificará el texto del botón "Abrir directorio" por "Ver imágenes". También se habilitará la opción de ordenar la caché por filas y se actualizará la imagen de bienvenida para hacerla más atractiva y profesional.

Para asegurar la compatibilidad, se adaptará la aplicación para Linux y Mac, y se realizarán pruebas unitarias para garantizar la calidad del software. Se creará un README personalizado que detalle la instalación y ejecución de la aplicación, y se reorganizarán los directorios, eliminando los que ya no se utilizan. Además, se subirá la release de la semana anterior, que incluye mejoras en el tratamiento de errores y la adaptación al inglés.

En cuanto a la documentación, se incluirán citas en la introducción de la memoria y se revisarán las correcciones en la misma. También se redactará el plan de proyecto software, la especificación de requisitos y la especificación de diseño en los anexos lo cual será la principal tarea de este sprint.

Finalmente, se realizará la release con todos los cambios implementados en este sprint, consolidando las mejoras de cara a la versión final.

## Resumen del tiempo empleado

A continuación se muestra un resumen del tiempo dedicado a la realización del proyecto.

Categoría	Tiempo (h)
<b>Sprint 0</b>	20
<b>Sprint 1</b>	32
<b>Sprint 2</b>	20
<b>Sprint 3</b>	26
<b>Sprint 4</b>	49
<b>Sprint 5</b>	54
<b>Sprint 6</b>	28
<b>Sprint 7</b>	42
<b>Sprint 8</b>	25+
<b>Total</b>	<b>296</b>

Tabla A.1: Desglose de tiempo dedicado al proyecto.

### A.3. Estudio de viabilidad

#### Viabilidad económica

En este apartado se analizarán los costes y beneficios que se estiman de haber sido desarrollado el proyecto en un entorno empresarial. He de mencionar que dichos datos son estimados.

#### Costes

##### Costes materiales:

En este apartado se desglosarán todos aquellos costes materiales, tanto hardware como software necesarios para el correcto desarrollo del proyecto. En ambos casos se considera que se van a amortizar a 5 años y se han utilizado durante los cuatro meses de desarrollo del proyecto.

El desglose de todos los dispositivos hardware necesarios para la realización del proyecto se encuentra a continuación en la figura A.2.

Concepto	Coste	Coste amortizado
Ordenador portátil	900,00€	60,00€
<b>Total</b>	<b>900,00€</b>	<b>60,00€</b>

Tabla A.2: Costes de hardware.

En cuanto a las licencias software no gratuitas se tiene el siguiente desglose en la figura A.3.

Concepto	Coste	Coste amortizado
Windows 10 Pro	259,00€	17,27€
MATLAB Academic (Anual)	262,00€	87,33€
<b>Total</b>	<b>521,00€</b>	<b>104,60€</b>

Tabla A.3: Costes de software.

Hay que destacar que la licencia de MATLAB utilizada ha sido la Academic que la propia Universidad de Burgos ofrece a los alumnos, la cual se paga anualmente.

##### Costes de personal:

El proyecto lo ha llevado a cabo un ingeniero informático recién graduado a tiempo completo durante cuatro meses. Se considera que el salario bruto

anual es de 24.000,00€ [5], el IRPF sobre el salario bruto es del 13,55 % y la Seguridad Social es del 6,35 % respectivamente como se puede ver en la figura A.4.

Concepto	Coste
Salario anual en bruto	24.000,00€
IRPF	3.251,04€
Seguridad Social	1.524,00€
Salario anual neto	19.224,96€
<b>Total 4 meses</b>	<b>8.000,00€</b>

Tabla A.4: Costes de personal.

### Costes totales:

El sumatorio de todos los distintos tipos de costes se encuentra en la figura A.5.

Concepto	Coste
Hardware	60,00€
Software	104,60€
Personal	8.000,00€
<b>Total</b>	<b>8.164,60€</b>

Tabla A.5: Costes totales.

### Beneficios

La aplicación desarrollada se distribuirá de forma libre y sin restricciones, permitiendo su uso sin limitaciones adicionales. La única condición necesaria para su funcionamiento será disponer de MATLAB. En una fase posterior, se evaluará la posibilidad de incorporar características adicionales que mejoren la experiencia del usuario, manteniendo siempre el enfoque en la accesibilidad y la libertad de uso.

### Viabilidad legal

Apartado dedicado al marco legal que envuelve el proceso de desarrollo del proyecto, su distribución, modificación y utilización. No es necesario atender al uso de licencias de terceros, pues todo el proyecto ha sido desarrollado con la licencia MATLAB Academic proporcionada por la Universidad de

Burgos para desarrolladores individuales o estudiantes y bajo una finalidad no comercial. En concreto, la licencia utilizada ha sido:

- **MATLAB Academic:** Al tratarse de un desarrollador individual, el uso de la licencia MATLAB Academic está justificado. Esta licencia es distribuida de forma gratuita por la Universidad de Burgos a sus alumnos.

El proyecto actual ha sido asignado bajo una licencia Creative Commons [1]. Esto implica que cualquier usuario es libre de compartir, redistribuir y adaptar la aplicación, siempre y cuando se cumplan las siguientes condiciones:

- **Atribución:** Se debe dar crédito apropiado al autor original, proporcionando un enlace a la licencia e indicando si se han realizado cambios.
- **NoComercial:** No se puede utilizar el material para fines comerciales.
- **CompartirIgual:** Si se remezcla, transforma, o crea a partir del material, se debe distribuir el material modificado bajo la misma licencia que el original.

Esta licencia asegura que la aplicación mantenga su accesibilidad y flexibilidad, fomentando la colaboración y el desarrollo comunitario mientras protege los derechos del autor original.

## Apéndice *B*

---

# Especificación de Requisitos

---

### B.1. Introducción

Este anexo recoge la especificación de requisitos que define el comportamiento del sistema desarrollado. Se han seguido las recomendaciones del estándar IEEE 830-1998 [2], que establece que una buena especificación de requisitos software debe ser:

- **Completa:** todos los requerimientos deben estar incluidos y todas las referencias definidas.
- **Consistente:** debe ser coherente tanto internamente como con otros documentos de especificación.
- **Inequívoca:** la redacción debe ser clara para evitar malentendidos.
- **Correcta:** el software debe cumplir con los requisitos especificados.
- **Trazable:** debe ser posible rastrear la historia, ubicación o aplicación de un ítem a través de su identificación documentada.
- **Priorizable:** los requisitos deben poder organizarse jerárquicamente según su relevancia para el negocio, clasificándolos en esenciales, condicionales y opcionales.
- **Modificable:** los requerimientos deben ser fácilmente modificables.
- **Verificable:** debe existir un método finito y sin costo para probar los requisitos.

## B.2. Objetivos generales

El proyecto tiene los siguientes objetivos generales.

- Demostrar que al aplicar métodos de transformación invariante, se consiguen mejores resultados a la hora de identificar piezas metálicas mediante métodos de segmentación de imágenes.
- Desarrollar una aplicación para de escritorio que permita al usuario identificar la pieza de una imagen que introduzca ofreciendo diferentes algoritmos tanto de transformación invariante (propuestos por Álvarez [3], Maddern [8], Krajník [7], Upcroft [13] y PCA [6]) como de segmentación de imágenes (K-Means [10], Fuzzy C-Means [9], GMM [11] y segmentación con información espacial [14]).
- Desarrollar una aplicación para de escritorio que permita mostrar la mejoría que suponen los métodos de transformación invariante mostrando una comparación de los resultados tanto visual como numérica.
- Guardar los resultados tanto de la imagen original, la resultante del método de transformación invariante como de las correspondientes imágenes tras la segmentación.

## B.3. Catálogo de requisitos

En este apartado se enumeran los requisitos específicos que se derivan de los objetivos generales del proyecto.

### Requisitos funcionales

- **RF-1 Gestión de transformación invariante:** la aplicación debe ser capaz de gestionar la transformación invariante sobre imágenes.
  - **RF-1.1 Seleccionar imagen:** el usuario debe poder seleccionar una imagen sobre la cual aplicar las diferentes transformaciones.
    - **RF-1.1.1 Imagen del sistema:** el usuario debe poder seleccionar una imagen de prueba contenida en el programa.
      - ◊ **RF-1.1.1.1 Carga automática de la imagen ground truth:** al haber seleccionado una imagen contenida en el programa, esta tendrá asociada una imagen ground truth.

- **RF-1.1.2 Imagen propia del usuario:** el usuario debe poder seleccionar una imagen propia. Al no tener imagen ground truth asociada esta no se cargará automáticamente.
  - ◊ **RF-1.1.2.1 Seleccionar una imagen ground truth propia:** debido a que la imagen que el usuario ha seleccionado es una imagen propia, el sistema permitirá aportar su correspondiente imagen ground truth.
  - ◊ **RF-1.1.2.2 Modificar imagen ground truth seleccionada:** el usuario debe poder modificar la imagen ground truth proporcionada en el caso de que se haya seleccionado una imagen propia.
- **RF-1.1.3 Modificar imagen seleccionada:** el usuario debe poder seleccionar una imagen diferente a la seleccionada anteriormente sobre la cual aplicar las diferentes transformaciones.
- **RF-1.2 Seleccionar algoritmo invariante:** el usuario debe poder seleccionar el algoritmo de transformación invariante que desee de la lista de algoritmos que la aplicación ofrece.
- **RF-1.3 Seleccionar algoritmo de agrupamiento:** el usuario debe poder seleccionar el algoritmo de agrupamiento invariante que desee de la lista de algoritmos que la aplicación ofrece.
- **RF-1.4 Seleccionar número de centros:** el usuario debe poder seleccionar el numero de centros que desee como parámetro del algoritmo de agrupamiento seleccionado dentro de unos limites preestablecidos.
- **RF-1.5 Almacenar resultados en caché:** la aplicación debe guardar en caché los resultados de la ejecución.
- **RF-1.6 Recuperar resultados almacenados en caché:** la aplicación recuperar los resultados de antiguas ejecuciones con los mismos parámetros para ahorrar costes computacionales.
- **RF-1.7 Guardar imágenes resultantes:** el usuario debe poder guardar los resultados de la ejecución en el directorio que desee.
- **RF-2 Gestión de caché:** la aplicación debe ser capaz de gestionar correctamente una memoria caché donde almacenar y mostrar los resultados de as ejecuciones del usuario.
  - **RF-2.1 Visualizar información de antiguas ejecuciones en formato tabla:** la aplicación debe mostrar en una tabla una

lista de los resultados de las distintas ejecuciones almacenadas en caché.

- **RF-2.1.1 Poder reordenar las columnas:** el usuario debe poder reordenar la tabla de la caché en función de los datos contenidos en las diferentes columnas.
  - **RF-2.1.2 Visualizar la imagen correspondiente a la fila seleccionada:** el usuario debe poder acceder en específico a la imagen a la que hace alusión cada fila de la tabla caché.
- **RF-2.2 Abrir el directorio de la caché:** el usuario debe poder acceder al directorio de la caché.
- **RF-2.3 Borrar contenido de la caché:** el usuario debe poder borrar el contenido de la caché.
- **RF-3 Configuración:** el usuario debe poder configurar los parámetros disponibles de la aplicación.
  - **RF-3.1 Seleccionar el idioma:** el usuario debe poder seleccionar el idioma de la aplicación dependiendo si desea trabajar con esta en español o en inglés.
- **RF-4 Ayuda:** el usuario debe poder obtener ayuda sobre los distintos elementos de la aplicación.
  - **RF-4.1 Documentación dentro de la aplicación:** la aplicación debe contener distintos apartados dedicados a documentar el funcionamiento y métodos aplicados en la aplicación.
  - **RF-4.2 Tooltips:** la aplicación debe proporcionar información sobre la funcionalidad de las distintas acciones dentro de esta.

## Requisitos no funcionales

- **RNF-1 Usabilidad:** la aplicación debe tener una interfaz de usuario intuitiva y fácil de usar, con una curva de aprendizaje mínima.
- **RNF-2 Mensajes de error explicativos:** la aplicación debe proporcionar mensajes de error claros y explicativos que ayuden al usuario a entender y resolver el problema.
- **RNF-3 Rendimiento:** la aplicación debe tener unos tiempos de carga aceptables.



- **RNF-4 Disponibilidad:** la aplicación debe estar preparada para funcionar en cualquier localización y sin acceso a internet.
- **RNF-5 Soporte:** la aplicación debe dar soporte a versiones de iguales o superiores a MATLAB 2023b.
- **RNF-6 Uso de recursos:** la aplicación debe ser eficiente en el uso de recursos del sistema, reutilizando los resultados de ejecuciones anteriores para reducir costes computacionales.
- **RNF-7 Modularidad:** la aplicación debe estar diseñada de manera modular, permitiendo la fácil adición o modificación de funcionalidades sin afectar otras partes del sistema.
- **RNF-8 Internacionalización:** la aplicación debe estar preparada para soportar varios idiomas.

## B.4. Especificación de requisitos

En esta sección se mostrará el diagrama de casos de uso resultante de los requisitos funcionales y se desarrollará cada uno de ellos

### Diagrama de casos de uso

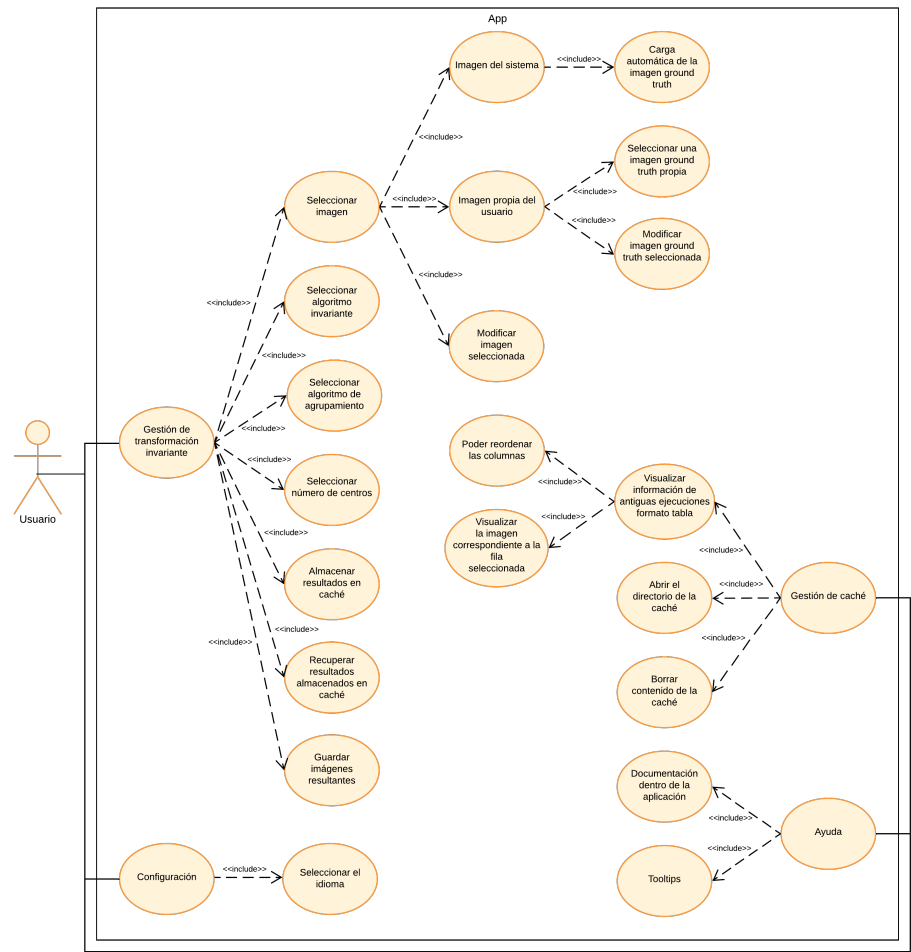


Figura B.1: Diagrama de casos de uso.

Actores

Casos de uso

Una muestra de cómo podría ser una tabla de casos de uso:

CU-1	Gestión de transformación invariante
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1, RF-1.1, RF-1.1.1, RF-1.1.1.1, RF-1.1.2, RF-1.1.2.1, RF-1.1.2.2, RF-1.1.3, RF-1.2, RF-1.3, RF-1.4, RF-1.5, RF-1.6, RF-1.7
<b>Descripción</b>	Permite al usuario aplicar sobre una imagen los distintos métodos de transformación invariante y guardar los resultados.
<b>Precondición</b>	La aplicación se encuentra disponible
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario pulsa en la pestaña de exploración de algoritmos</li> <li>2. El usuario selecciona una imagen de pieza metálica.</li> <li>3. El usuario selecciona un algoritmo de transformación invariante.</li> <li>4. El usuario selecciona un algoritmo de agrupamiento.</li> <li>5. El usuario selecciona un numero de centros (por defecto 2).</li> <li>6. El usuario hace click en el botón de ejecutar.</li> <li>7. El programa ejecuta sobre la imagen seleccionada el algoritmo invariante seleccionado y mas tarde el algoritmo de agrupamiento sobre la original y sobre la invariante.</li> <li>8. En el caso de que tenga imagen ground truth calcula el porcentaje de acierto.</li> <li>9. El programa guarda en la cache los resultados.</li> <li>10. El programa muestra cuatro imágenes tras finalizarse la ejecución: <ul style="list-style-type: none"> <li>■ Imagen original.</li> <li>■ Imagen original segmentada.</li> <li>■ Imagen invariante.</li> <li>■ Imagen invariante segmentada.</li> </ul> </li> <li>11. El usuario hace click en el botón de guardar resultados y selecciona un directorio donde guardar las imágenes.</li> </ol>
<b>Postcondición</b>	Se muestran los resultados de la transformación invariante y las segmentaciones indicando en el caso de que haya imagen ground truth la tasa de acierto.
<b>Excepciones</b>	Error (mensaje)
<b>Importancia</b>	Alta

Tabla B.1: CU-1 Gestión de transformación invariante.

CU-2	Seleccionar imagen
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.1, RF-1.1.1, RF-1.1.1.1, RF-1.1.2, RF-1.1.2.1, RF-1.1.2.2, RF-1.1.3
<b>Descripción</b>	Permite al usuario seleccionar una imagen sobre la que aplicar los distintos métodos de transformación invariante.
<b>Precondición</b>	La aplicación se encuentra disponible
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario pulsa en la pestaña de exploración de algoritmos</li> <li>2. El usuario hace click en el botón de cargar imagen de pieza metálica.</li> <li>3. Se muestra el directorio de imágenes del programa.</li> <li>4. El usuario selecciona una imagen del programa o propia.</li> <li>5. El usuario hace click en el botón de aceptar. <ul style="list-style-type: none"> <li>■ En el caso de seleccionar una imagen del programa, la aplicación busca su correspondiente imagen ground truth.</li> <li>■ En el caso de seleccionar una imagen propia, se notifica al usuario de que al no ser una imagen del programa esta no tiene imagen ground truth asociada.</li> </ul> </li> <li>6. Si no hay ningún error: se guardan en memoria la imagen y se muestra en sus correspondiente plot y si es una imagen del programa también la imagen ground truth.</li> </ol>
<b>Postcondición</b>	<ol style="list-style-type: none"> <li>1. La imagen seleccionada se ha cargado correctamente.</li> <li>2. Activa la interacción de los selectores de método invariante, método de agrupamiento, numero de centro y el botón de ejecutar.</li> </ol>
<b>Excepciones</b>	Se ha cancelado la selección de imágenes. (mensaje). Image selection has been canceled. (mensaje). Imagen ground truth no encontrada. (mensaje). Ground truth image not found (mensaje).
<b>Importancia</b>	Alta

Tabla B.2: CU-2 Seleccionar imagen.

CU-3	Imagen del sistema
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.1.1, RF-1.1.1.1
<b>Descripción</b>	El usuario debe poder seleccionar una imagen de prueba contenida en el programa
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha hecho click en el botón de cargar imagen de pieza metálica.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona una imagen de las proporcionadas por el sistema.</li> <li>2. La aplicación carga la imagen seleccionada.</li> <li>3. La aplicación busca la imagen ground truth correspondiente a la imagen seleccionada.</li> <li>4. Si no hay ningún error se muestra la imagen y su correspondiente imagen ground truth.</li> </ol>
<b>Postcondición</b>	La imagen seleccionada se ha cargado correctamente.
<b>Excepciones</b>	Se ha cancelado la selección de imágenes. (mensaje). Image selection has been canceled. (mensaje). Imagen ground truth no encontrada. (mensaje). Ground truth image not found (mensaje).
<b>Importancia</b>	Alta

Tabla B.3: CU-3 Imagen del sistema.

CU-4	Carga automática de la imagen ground truth
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.1.1.1
<b>Descripción</b>	Al haber seleccionado una imagen de la aplicación, se cargara su correspondiente imagen ground truth.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha hecho click en el botón de cargar imagen de pieza metálica. El usuario ha seleccionado una imagen del sistema.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. La aplicación busca la correspondiente imagen ground truth de la imagen seleccionada en el directorio de imágenes ground truth.</li> <li>2. Si no hay ningún error se muestra la imagen ground truth en el plot correspondiente.</li> </ol>
<b>Postcondición</b>	Imagen ground truth se ha cargado correctamente.
<b>Excepciones</b>	Imagen ground truth no encontrada. (mensaje). Ground truth image not found (mensaje).
<b>Importancia</b>	Media.

Tabla B.4: CU-4 Carga automática de la imagen ground truth.

CU-5	Imagen propia del usuario
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.1.2, RF-1.1.2.1, RF-1.1.2.2
<b>Descripción</b>	El usuario debe poder seleccionar una imagen propia.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha hecho click en el botón de cargar imagen de pieza metálica.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona una imagen propia.</li> <li>2. La aplicación carga la imagen seleccionada.</li> <li>3. La aplicación busca informa al usuario que al ser una imagen propia, esta no tiene imagen ground truth asociada.</li> <li>4. Si no hay ningún error se muestra la imagen.</li> <li>5. La aplicación activa la interacción del botón de cargar una imagen ground truth propia.</li> </ol>
<b>Postcondición</b>	La imagen seleccionada se ha cargado correctamente.
<b>Excepciones</b>	Se ha cancelado la selección de imágenes. (mensaje). Image selection has been canceled. (mensaje). Imagen ground truth no encontrada. (mensaje). Ground truth image not found (mensaje).
<b>Importancia</b>	Alta

Tabla B.5: CU-5 Imagen propia del usuario.

CU-6	Seleccionar una imagen ground truth propia
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.1.2.1
<b>Descripción</b>	La descripción del CU
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen propia.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el botón de cargar imagen ground truth.</li> <li>2. La aplicación abre un directorio para que el usuario seleccione el archivo que desee.</li> <li>3. El usuario selecciona la imagen que quiere utilizar como ground truth.</li> <li>4. Si no hay ningún error se muestra la imagen ground truth.</li> </ol>
<b>Postcondición</b>	Imagen ground truth se ha cargado correctamente.
<b>Excepciones</b>	Error al leer la imagen ground truth: (mensaje). Error reading ground truth image: (mensaje).
<b>Importancia</b>	Media

Tabla B.6: CU-6 Seleccionar una imagen ground truth propia.



CU-7	Modificar imagen ground truth seleccionada
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.1.2.2
<b>Descripción</b>	El usuario debe poder modificar la imagen ground truth proporcionada en el caso de que se haya seleccionado una imagen propia.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen propia. El usuario ha cargado una imagen ground truth propia.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el botón de cargar imagen ground truth.</li> <li>2. La aplicación abre un directorio para que el usuario seleccione el archivo que desee.</li> <li>3. El usuario selecciona la imagen que quiere utilizar como ground truth.</li> <li>4. La aplicación elimina de memoria la imagen ground truth anterior y guarda en su lugar la nueva.</li> <li>5. Si no hay ningún error se muestra la imagen ground truth sustituyendo a la anterior.</li> </ol>
<b>Postcondición</b>	Imagen ground truth nueva se ha cargado correctamente.
<b>Excepciones</b>	Error al leer la imagen ground truth: (mensaje). Error reading ground truth image: (mensaje).
<b>Importancia</b>	Media

Tabla B.7: CU-7 Modificar imagen ground truth seleccionada.

CU-8	Modificar imagen seleccionada
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.1.3
<b>Descripción</b>	El usuario debe poder seleccionar una imagen diferente a la seleccionada anteriormente
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen anteriormente.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el botón de cargar imagen de pieza metálica.</li> <li>2. Se muestra el directorio de imágenes del programa.</li> <li>3. El usuario selecciona una imagen del programa o propia.</li> <li>4. El usuario hace click en el botón de aceptar. <ul style="list-style-type: none"> <li>■ En el caso de seleccionar una imagen del programa, la aplicación busca su correspondiente imagen ground truth.</li> <li>■ En el caso de seleccionar una imagen propia, se notifica al usuario de que al no ser una imagen del programa esta no tiene imagen ground truth asociada.</li> </ul> </li> <li>5. Si no hay ningún error se dejan de mostrar las imágenes tanto de entrada como de salida de anteriores ejecuciones, se boquea la interacción del botón de guardar resultados y: <ul style="list-style-type: none"> <li>■ En el caso de seleccionar una imagen del programa, se guardan en memoria tanto la imagen como la imagen ground truth y ambas se muestran en sus respectivos plots.</li> <li>■ En el caso de seleccionar una imagen propia, se guardan en memoria la imagen y se muestra en sus correspondiente plot.</li> </ul> </li> </ol>
<b>Postcondición</b>	La imagen seleccionada se ha cargado correctamente.
<b>Excepciones</b>	Se ha cancelado la selección de imágenes. (mensaje). Image selection has been canceled. (mensaje). Imagen ground truth no encontrada. (mensaje). Ground truth image not found (mensaje).
<b>Importancia</b>	Alta

Tabla B.8: CU-8 Modificar imagen seleccionada.

<b>CU-9</b>	<b>Seleccionar algoritmo invariante</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.2
<b>Descripción</b>	El usuario debe poder seleccionar el algoritmo de transformación invariante que desee de la lista de algoritmos que la aplicación ofrece.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en una de las distintas opciones del contenedor de algoritmos invariantes.</li> <li>2. Si no hay ningún error, se guarda la preferencia del usuario.</li> </ol>
<b>Postcondición</b>	Se ha seleccionado correctamente el algoritmo invariante deseado.
<b>Excepciones</b>	Error (mensaje).
<b>Importancia</b>	Alta

Tabla B.9: CU-9 Seleccionar algoritmo invariante.

CU-10	Seleccionar algoritmo de agrupamiento
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.3
<b>Descripción</b>	El usuario debe poder seleccionar el algoritmo de agrupamiento invariante que desee de la lista de algoritmos que la aplicación ofrece.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en una de las distintas opciones del contenedor de algoritmos de agrupamiento.</li> <li>2. Si no hay ningún error, se guarda la preferencia del usuario.</li> </ol>
<b>Postcondición</b>	Se ha seleccionado correctamente el algoritmo de agrupamiento deseado.
<b>Excepciones</b>	Error (mensaje).
<b>Importancia</b>	Alta

Tabla B.10: CU-10 Seleccionar algoritmo de agrupamiento.

CU-11	Seleccionar número de centros
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.4
<b>Descripción</b>	El usuario debe poder seleccionar el numero de centros que desee como parámetro del algoritmo de agrupamiento seleccionado dentro de unos limites preestablecidos.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el selector del numero de centros.</li> <li>2. El usuario escribe el numero de centros deseado.</li> <li>3. Si no hay ningún error, se guarda la preferencia del usuario.</li> </ol>
<b>Postcondición</b>	Se ha seleccionado correctamente la cantidad de centros deseada.
<b>Excepciones</b>	Value must be numeric (mensaje). Value must be between 2 and Infinity (mensaje).
<b>Importancia</b>	Alta

Tabla B.11: CU-11 Seleccionar número de centros.

CU-12	Almacenar resultados en caché
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.5
<b>Descripción</b>	La aplicación debe guardar en caché los resultados de la ejecución.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen. El usuario ha pulsado el botón de ejecutar.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El programa ejecuta el algoritmo de transformación invariante sobre la imagen seleccionada.</li> <li>2. El programa aplica el algoritmo sobre la imagen original y sobre la imagen invariante.</li> <li>3. El programa informa al usuario de que se va a proceder a guardar la ejecución en la memoria caché.</li> <li>4. El programa guarda las tres imágenes en el directorio caché añadiendo el acierto en caso haber propiciado una imagen ground truth.</li> <li>5. El programa actualiza la tabla de la memoria caché con la información de las tres imágenes.</li> </ol>
<b>Postcondición</b>	Se han guardado correctamente las tres imágenes resultantes de la ejecución en el directorio caché. Se ha actualizado la tabla de la memoria caché.
<b>Excepciones</b>	Error (mensaje).
<b>Importancia</b>	Media

Tabla B.12: CU-12 Almacenar resultados en caché.

CU-13	Recuperar resultados almacenados en caché
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.6
<b>Descripción</b>	La aplicación recuperar los resultados de antiguas ejecuciones con los mismos parámetros para ahorrar costes computacionales.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen. El usuario ha pulsado el botón de ejecutar.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. La aplicación comprueba el la cache si hay alguna ejecución de la misma imagen con los mismos parámetros.</li> <li>2. La aplicación recupera las imágenes que coincidan, estas puede ser: <ul style="list-style-type: none"> <li>■ La imagen invariante si coincide únicamente el método invariante sobre esa imagen.</li> <li>■ La imagen original segmentada si coinciden únicamente el método de agrupamiento y la cantidad de centros sobre esa imagen.</li> <li>■ La imagen invariante, la imagen original segmentada y la imagen invariante segmentada si coinciden todos los parámetros.</li> </ul> </li> </ol>
<b>Postcondición</b>	Se ha recuperado una o más imágenes si ha habido una ejecución sobre la misma imagen anterior sin la necesidad de ejecutar de nuevo.
<b>Excepciones</b>	Error (mensaje).
<b>Importancia</b>	Media

Tabla B.13: CU-13 Recuperar resultados almacenados en caché.

CU-14	Guardar imágenes resultantes
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-1.7
<b>Descripción</b>	El usuario debe poder guardar los resultados de la ejecución en el directorio que desee.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario ha cargado una imagen. El usuario ejecutado el algoritmo sobre la imagen. La aplicación ha mostrado los resultados de la ejecución.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el botón de guardar resultados.</li> <li>2. El usuario selecciona el directorio donde quiere guardar las imágenes.</li> <li>3. La aplicación guarda las imágenes con nombres descriptivos que indiquen con que parámetros se ha ejecutado.</li> </ol>
<b>Postcondición</b>	Se han guardado correctamente las tres imágenes resultantes de la ejecución en el directorio seleccionado.
<b>Excepciones</b>	Error al guardar imágenes: (mensaje). Error saving images: (mensaje).
<b>Importancia</b>	Media

Tabla B.14: CU-14 Guardar imágenes resultantes.



CU-15	Gestión de caché
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-2, RF-2.1, RF-2.1.1, RF-2.1.2, RF-2.2, RF-2.3
<b>Descripción</b>	La aplicación debe ser capaz de gestionar correctamente una memoria caché donde almacenar y mostrar los resultados de las ejecuciones del usuario.
<b>Precondición</b>	La aplicación se encuentra disponible.
<b>Acciones</b>	<ol style="list-style-type: none"><li>1. La aplicación al iniciarse crea en caso de no existir un directorio que se utilizara como memoria caché.</li></ol>
<b>Postcondición</b>	Se han creado/cargado correctamente el directorio que se usará como memoria caché.
<b>Excepciones</b>	Error al crear el directorio de la memoria caché (mensaje).
<b>Importancia</b>	Media

Tabla B.15: CU-15 Gestión de caché.

<b>CU-16</b>	<b>Visualizar información de antiguas ejecuciones en formato tabla</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-2.1, RF-2.1.1, RF-2.1.2
<b>Descripción</b>	La aplicación debe mostrar en una tabla una lista de los resultados de las distintas ejecuciones almacenadas en caché.
<b>Precondición</b>	La aplicación se encuentra disponible. La aplicación cuenta con el directorio de la caché.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el menú de histórico de exploraciones.</li> <li>2. La aplicación lee una por una cada imagen almacenada en el directorio de la caché.</li> <li>3. La aplicación identifica los campos de las imágenes y en caso de tener la estructura adecuada añade una nueva fila a la tabla con la información de la imagen. <ul style="list-style-type: none"> <li>■ Esto se consigue mediante expresiones regulares.</li> </ul> </li> <li>4. La aplicación tras cada ejecución guarda la información de cada imagen resultante en un nueva fila en la tabla de la caché.</li> </ol>
<b>Postcondición</b>	La tabla de la memoria caché se muestra correctamente y esta actualizada.
<b>Excepciones</b>	Error al mostrar caché (mensaje).
<b>Importancia</b>	Media

Tabla B.16: CU-16 Visualizar información de antiguas ejecuciones en formato tabla.

CU-17	Poder reordenar las columnas
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-2.1.1
<b>Descripción</b>	El usuario debe poder reordenar la tabla de la caché en función de los datos contenidos en las diferentes columnas.
<b>Precondición</b>	La aplicación se encuentra disponible. La aplicación cuenta con al menos una imagen en la caché.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el icono de ordenar por la fila seleccionada que se encuentra a la derecha.</li> <li>2. La aplicación depende del numero de clicks de usuario reordena la tabla en función de esa columna en orden: <ul style="list-style-type: none"> <li>■ Ascendente.</li> <li>■ Descendente.</li> <li>■ Elimina la reordenación.</li> </ul> </li> <li>3. Si no hay ningún problema, la tabla se reordena según la selección del usuario.</li> </ol>
<b>Postcondición</b>	La tabla de la memoria caché se muestra reordenada correctamente.
<b>Excepciones</b>	Error al ordenar por columna (mensaje). No hay imágenes guardadas en la caché. (mensaje).
<b>Importancia</b>	Baja

Tabla B.17: CU-17 Poder reordenar las columnas.

<b>CU-18</b>	<b>Visualizar la imagen correspondiente a la fila seleccionada</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-2.1.2
<b>Descripción</b>	El usuario debe poder acceder en específico a la imagen a la que hace alusión cada fila de la tabla caché.
<b>Precondición</b>	La aplicación se encuentra disponible. La aplicación cuenta con al menos una imagen en la caché.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en cualquiera de los iconos de la última columna de la imagen.</li> <li>2. La aplicación ha de abrir en una ventana a parte la imagen correspondiente a la fila mediante expresiones regulares.</li> <li>3. La aplicación ha de facilitar una serie de opciones como guardar la imagen o imprimirla.</li> </ol>
<b>Postcondición</b>	La imagen correspondiente a la fila seleccionada se muestra correctamente.
<b>Excepciones</b>	El archivo de imagen: (imagen)(extensión) (mensaje). The image file: (imagen)(extensión) (mensaje).
<b>Importancia</b>	Media

Tabla B.18: CU-18 Visualizar la imagen correspondiente a la fila seleccionada.

CU-19	Abrir el directorio de la caché
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-2.2
<b>Descripción</b>	El usuario debe poder acceder al directorio de la caché.
<b>Precondición</b>	La aplicación se encuentra disponible.
<b>Acciones</b>	<ol style="list-style-type: none"><li>1. El usuario hace click en el botón de ver imágenes.</li><li>2. La aplicación ha de abrir el directorio correspondiente a la memoria caché.</li></ol>
<b>Postcondición</b>	El usuario ha de visualizar en pantalla el directorio de la memoria caché.
<b>Excepciones</b>	Error al intentar abrir la carpeta de caché: (mensaje). Error while trying to open the cache folder: (mensaje).
<b>Importancia</b>	Baja

Tabla B.19: CU-19 Abrir el directorio de la caché.

<b>CU-20</b>	<b>Borrar contenido de la caché</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-2.3
<b>Descripción</b>	El usuario debe poder borrar el contenido de la caché.
<b>Precondición</b>	La aplicación se encuentra disponible. La aplicación cuenta con al menos una imagen en la caché.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el botón de borrar datos.</li> <li>2. La aplicación mediante un menú de confirmación advierte de que de continuar, se borrarán todos los datos de la memoria caché.</li> <li>3. El usuario hace click en el botón de confirmar.</li> <li>4. La aplicación borra el contenido del directorio de la memoria caché.</li> <li>5. La aplicación actualiza el estado de la tabla que muestra las imágenes guardadas en la memoria cache, esta pasara a estar vacía.</li> <li>6. La aplicacion informa al usuario de que se han borrado todos los datos de la memoria cache.</li> </ol>
<b>Postcondición</b>	El contenido de la memoria caché se ha borrado correctamente.
<b>Excepciones</b>	Error al intentar borrar la caché: (mensaje). Error while trying to clear the cache: (mensaje).
<b>Importancia</b>	Baja

Tabla B.20: CU-20 Borrar contenido de la caché.

CU-21	Configuración
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-3, RF-3.1
<b>Descripción</b>	El usuario debe poder configurar los parámetros disponibles de la aplicación.
<b>Precondición</b>	La aplicación se encuentra disponible.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario ha de hacer click en la opción u opciones que la aplicación ofrezca.</li> <li>2. La aplicación ha de guardar la selección del usuario y realizar los ajustes pertinentes.</li> <li>3. Si no hay errores, el usuario verá los cambios reflejados.</li> </ol>
<b>Postcondición</b>	El usuario ha de poder aplicar la configuración deseada.
<b>Excepciones</b>	Error al cambiar el idioma: (mensaje). Error while changing the language: (mensaje).
<b>Importancia</b>	Media

Tabla B.21: CU-21 Configuración.

<b>CU-22</b>	<b>Seleccionar el idioma</b>
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-3.1
<b>Descripción</b>	El usuario debe poder seleccionar el idioma de la aplicación dependiendo si desea trabajar con esta en español o en inglés.
<b>Precondición</b>	La aplicación se encuentra disponible.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. Por defecto la aplicación estará en español y con el selector a la izquierda (ES).</li> <li>2. El usuario debe hacer click en el selector de idioma, este cambiara de posición y con ello el idioma de la aplicación. <ul style="list-style-type: none"> <li>■ Al hacer click este pasara a estar a la derecha (EN).</li> <li>■ Al hacer click por segunda vez este pasara a estar a la izquierda (ES).</li> </ul> </li> <li>3. Si no hay errores, el usuario verá la aplicación en el idioma seleccionado.</li> </ol>
<b>Postcondición</b>	El usuario ha de ver la aplicación al completo en el idioma seleccionado.
<b>Excepciones</b>	Error al cambiar el idioma: (mensaje). Error while changing the language: (mensaje).
<b>Importancia</b>	Media

Tabla B.22: CU-22 Seleccionar el idioma.



CU-23	Ayuda
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-4, RF-4.1, RF-4.2
<b>Descripción</b>	El usuario debe poder obtener ayuda sobre los distintos elementos de la aplicación.
<b>Precondición</b>	La aplicación se encuentra disponible.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario hace click en el menú de ayuda.</li> <li>2. El usuario selecciona dentro del menú de ayuda el área de interés.</li> <li>3. La aplicación le muestra una explicación detallada del funcionamiento, algoritmo o proceso seleccionado.</li> <li>4. El usuario hace click en el menú de exploración de algoritmos o en el de histórico de exploraciones.</li> <li>5. El usuario mantiene el puntero dos segundos sobre cualquier botón de la aplicación sobre el cual quiera obtener más información.</li> <li>6. La aplicación muestra una pequeña ventana con información relevante.</li> </ol>
<b>Postcondición</b>	El usuario ha de poder solucionar cualquier duda que le surja sobre la funcionalidad o el funcionamiento de la aplicación mediante la interfaz de esta.
<b>Excepciones</b>	Error al cargar liveness (mensaje). Error al cargar tooltips (mensaje).
<b>Importancia</b>	Alta

Tabla B.23: CU-23 Ayuda.

CU-24	Documentación dentro de la aplicación
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-4.1
<b>Descripción</b>	La aplicación debe contener distintos apartados dedicados a documentar el funcionamiento y métodos aplicados en la aplicación.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario hace click en el menú de ayuda.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona dentro del menú de ayuda el área de interés.</li> <li>2. La aplicación le muestra una explicación detallada del funcionamiento, algoritmo o proceso seleccionado.</li> </ol>
<b>Postcondición</b>	El usuario ha de poder entender mediante la documentación facilitada el funcionamiento de cada apartado y algoritmo dentro de la aplicación.
<b>Excepciones</b>	Error al cargar livescripts (mensaje).
<b>Importancia</b>	Alta

Tabla B.24: CU-24 Documentación dentro de la aplicación.

CU-25	Tooltips
<b>Versión</b>	1.0
<b>Autor</b>	Jonás Martínez Sanllorente
<b>Requisitos asociados</b>	RF-4.2
<b>Descripción</b>	La aplicación debe proporcionar información sobre la funcionalidad de las distintas acciones dentro de esta.
<b>Precondición</b>	La aplicación se encuentra disponible. El usuario hace click en el menú de exploración de algoritmos o en el de histórico de exploraciones.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario mantiene el puntero dos segundos sobre cualquier botón de la aplicación sobre el cual quiera obtener más información.</li> <li>2. La aplicación muestra una pequeña ventana con información relevante.</li> </ol>
<b>Postcondición</b>	La aplicación ha de facilitar información sobre la utilidad de los distintos botones y sobre las posibles restricciones de algunos campos.
<b>Excepciones</b>	Error al cargar tooltips (mensaje).
<b>Importancia</b>	Media

Tabla B.25: CU-25 Tooltips.



## *Apéndice C*

---

# **Especificación de diseño**

---

### **C.1. Introducción**

En este apéndice se abordarán los aspectos relacionados con el diseño de tratamiento de datos, el diseño de procesos y el diseño arquitectónico, los cuales se siguieron durante el desarrollo del proyecto.

### **C.2. Diseño de datos**

### **C.3. Diseño procedimental**

En este apartado se recogen los detalles más relevantes respecto a la ejecución de la aplicación desde que el usuario selecciona una imagen a analizar hasta que guarda los resultados.

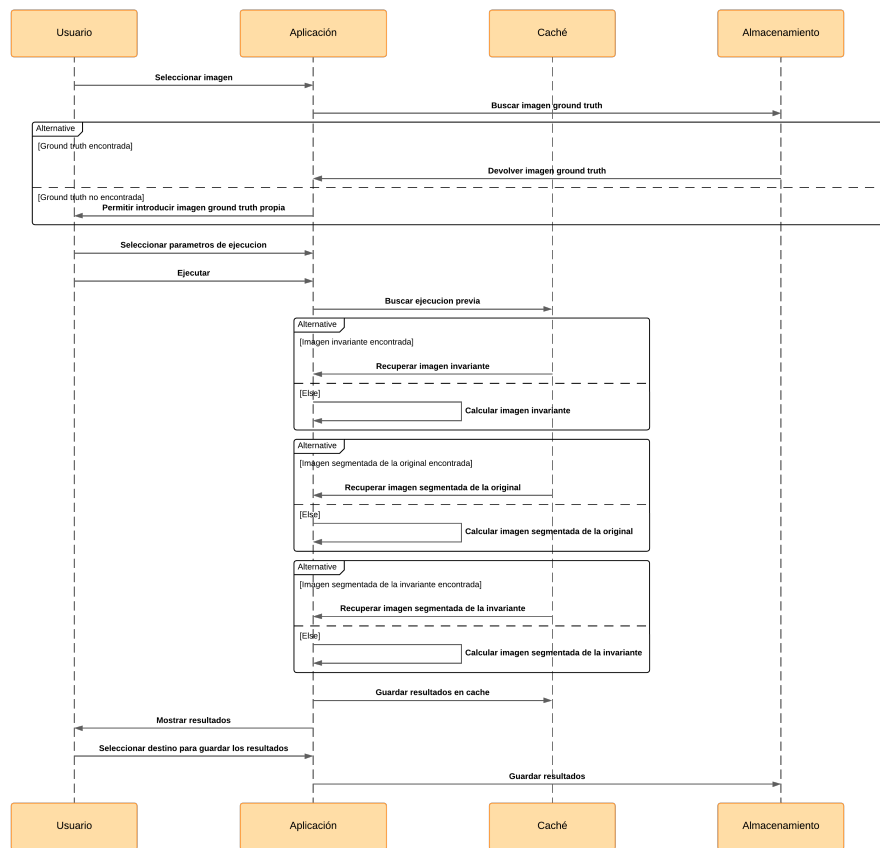


Figura C.1: Diagrama de secuencia de los distintos componentes de la aplicación.

## C.4. Diseño arquitectónico

## *Apéndice D*

---

# Documentación técnica de programación

---

## D.1. Introducción

En este anexo se detalla la documentación técnica de programación, incluyendo la instalación del entorno de desarrollo, la estructura de la aplicación, su compilación, la configuración de los diversos servicios de integración utilizados y las baterías de pruebas realizadas.

## D.2. Estructura de directorios

El repositorio del proyecto tiene la siguiente estructura:

- **/:** directorio raíz del repositorio. Es este se encuentra todos los directorios del proyecto además de una README.
- **/app:** directorio de la aplicación. Este contiene tanto la propia aplicación como aquellos directorios necesarios para la correcta ejecución de la misma.
- **/app/src:** directorio del código fuente de la aplicación.
- **/app/src/functions:** directorio de los scripts utilizados por el sistema.

- **/app/src/livescripts:** directorio de los livescripts y sus correspondientes HTML de la documentación de dentro de la aplicación tanto en español como en inglés.
- **/app/src/livescripts/img:** directorio de las imágenes utilizadas en la documentación de los livescripts.
- **/app/src/test:** directorio de los tests unitarios de los scripts de ‘/functions’. Cuentan con un README para poderlos ejecutar correctamente.
- **/app/data:** directorio de las imágenes proporcionadas y sus respectivas imágenes ground truth, además contiene un README para explicar la nomenclatura utilizada en las imágenes.
- **/app/data/Imagenes:** directorio de las imágenes proporcionadas por la aplicación.
- **/app/data/Imagenes\_ground\_truth:** directorio de las imágenes ground truth correspondientes a las imágenes proporcionadas por la aplicación.
- **/app/cache:** directorio de la memoria caché de la aplicación. En el se guardarán las imágenes generadas durante la ejecución.
- **/app/results:** directorio donde se almacenarán las imágenes por defecto si el usuario decide guardar los resultados de la ejecución.
- **/docs:** directorio de la documentación del proyecto.
- **/docs/img:** directorio de las imágenes utilizadas en la documentación.
- **/docs/tex:** directorio de las fuentes en formato  $\text{\LaTeX}$  para memoria y anexos.

### D.3. Manual del programador

El siguiente manual tiene como objetivo explicar cómo un programador puede descargar el proyecto y desarrollar su propia versión.



## Entorno de desarrollo

Para poder trabajar con el proyecto es necesario tener instalado esta serie de programas y dependencias.

- MATLAB 2023b.
- MATLAB Add-On Toolboxes.
- GitHub Desktop.

A continuación, se detallan las instrucciones para instalar y configurar correctamente cada uno de ellos.

### MATLAB 2023b

El principal e indispensable programa para la ejecución de este proyecto es MATLAB, mas concretamente la versión 2023b. Para descargarlo tenemos que ir a la página oficial de MATLAB [12]. Una vez iniciada la aplicación nos pedirá introducir una cuenta de MATLAB, la Universidad de Burgos ofrece a los alumnos el acceso con su correo institucional a la versión Academic de MATLAB, versión con la cual se ha realizado este proyecto.

### MATLAB Add-On Toolboxes

Una vez instalado MATLAB tendremos que instalar una serie de Add-On Toolboxes necesarias para la correcta ejecución del proyecto. Estas son las siguientes:

- Statistics and Machine Learning Toolbox.
- Fuzzy Logic Toolbox.
- HMRF-EM-image.
- MATLAB Compiler.

Esto lo podemos hacer desde el menú superior ‘APPS’ y tras esto pulsando en ‘Get More Apps’. Tras esto, se nos mostrara una nueva ventana donde mediante un buscador podremos acceder a las Add-On Toolboxes necesarias. A continuación, se muestra un ejemplo de la instalación de ‘Statistics and Machine Learning Toolbox’ en la figura D.1.

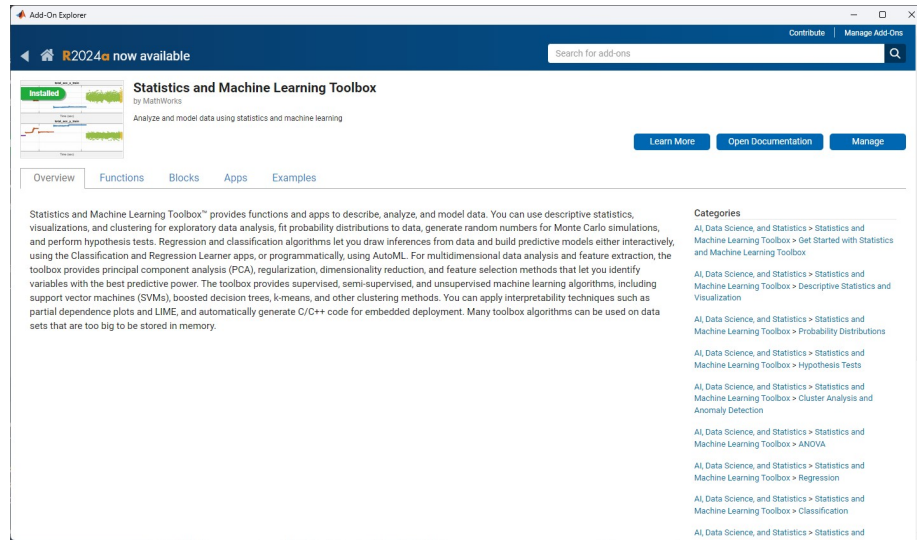


Figura D.1: Instalación Statistics and Machine Learning Toolbox

## GitHub Desktop

La instalación de GitHub Desktop no es estrictamente necesaria pero si recomendable ya que permite tener un control de versiones lo cual es de gran interés a la hora de hacer cambios en la aplicación.

Para instalar GitHub Desktop tendremos que ir a la pagina oficial [4] y desde allí descargar los archivos necesarios. Tras la instalación nos pedirá nuestro usuario de GitHub.

## Descarga del repositorio

Para la descarga del repositorio hay dos principales opciones:

### Obtención manual del código fuente

En el caso de que no estimemos necesario un control de versiones, podemos descargar directamente el archivo .zip del proyecto desde el repositorio de GitHub (<https://github.com/jms1008/TFG-Invariantes-de-iluminacion-en-piezas-metalicas>). Esto se realiza pulsando en el botón 'Code' y mas tarde en el de 'Download ZIP' como se puede ver en la figura D.2:

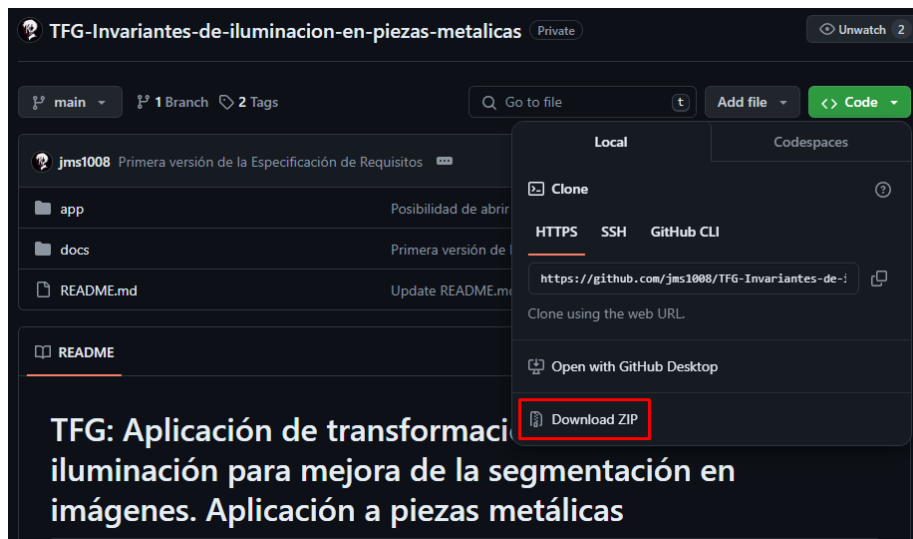


Figura D.2: Descarga del repositorio mediante el entorno web.

Tras esto solo quedaría descomprimir el proyecto en el directorio deseado.

### Obtención del código fuente mediante GitHub Desktop

En el caso de querer tener un control de versiones, podemos utilizar la aplicación GitHub Desktop mencionada anteriormente en D.3.

Para ello debemos entrar a la aplicación pulsamos en clonar repositorio e introducimos el enlace del proyecto que queremos clonar, en este caso el enlace del repositorio de este proyecto es el siguiente <https://github.com/jms1008/TFG-Invariantes-de-iluminacion-en-piezas-metalicas>. A continuación indicamos donde queremos almacenar el repositorio.

### Importación del proyecto

Para la correcta ejecución del proyecto hemos de abrir MATLAB 2023b e ir dentro de MATLAB al directorio donde tengamos el repositorio. Una vez allí, tendremos que escribir en la consola 'appdesigner' para que se nos abra la ventana correspondiente donde podremos visualizar, modificar y ejecutar entre muchas otras funcionalidades la aplicación.

## MATLAB App Designer

La en la primera pantalla que aparece nada mas abrir MATLAB App Designer nos permitirá abrir un proyecto. Aquí hemos de seleccionar el archivo ‘InvIPM.mlapp’ que se encuentra en el directorio ‘/app’.

Una vez abierto el proyecto veremos una vista de diseño de la aplicación donde podremos modificar la posición y algunos valores dentro del apartado visual como se muestra en la figura D.3 y otro botón que nos permite acceder a la vista de código.

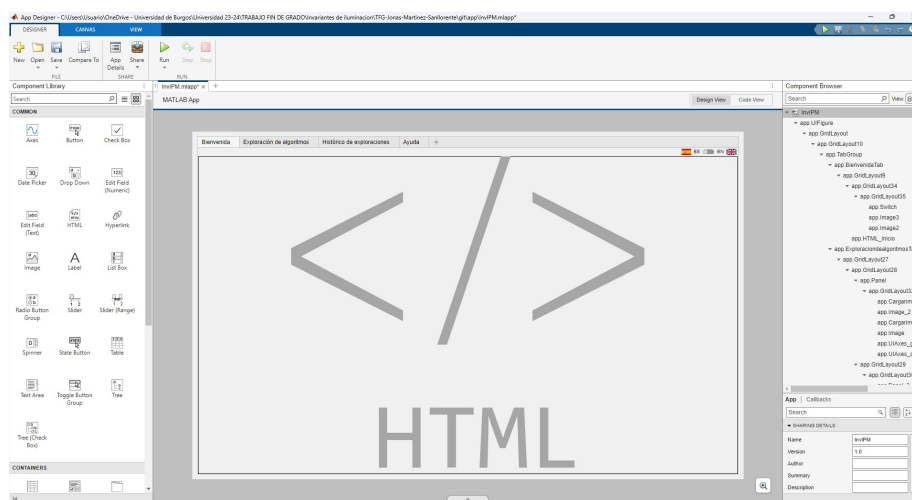


Figura D.3: Ventana principal de MATLAB App Designer.

Es en la vista de código donde podremos modificar mas en profundidad los objetos visuales y programar las funcionalidades de las distintas interacciones de la aplicación.

Aunque la forma de programar en MATLAB App Designer es en si muy parecida a como es en MATLAB, hay una serie de cambios y funcionalidades las cuales soy muy relevantes. En el siguiente enlace se encuentra una guía de las funcionalidades y la forma de programar en MATLAB App Designer, <https://es.mathworks.com/help/matlab/app-designer.html>

## D.4. Compilación, instalación y ejecución del proyecto

A continuación se va a explicar como:

- Compilar y ejecutar el proyecto dentro de MATLAB.
- Crear una nueva release mediante una MATLAB Standalone Application.
- Directorios creados por el compilador de MATLAB.
- Ejecución de la Standalone App.

## Compilación en App Designer

MATLAB permite ejecutar directamente desde la interfaz compilar y ejecutar la aplicación. Para ello simplemente tendremos que pulsar el icono de ‘run’ que se encuentra en la barra superior, dicho icono es el que se muestra en la figura D.4.

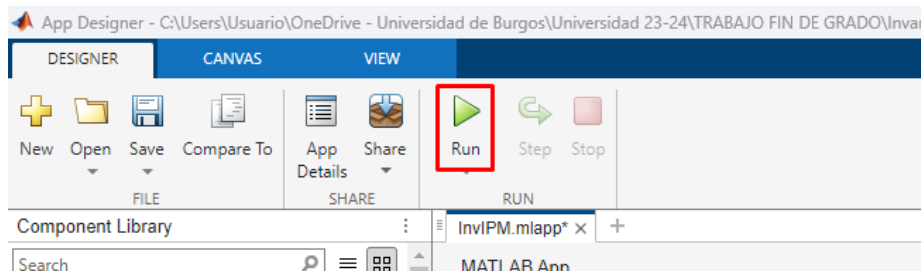


Figura D.4: Icono de compilación y ejecución de MATLAB App Designer.

Tras esto se nos abrirá la aplicación en una ventana emergente donde podremos probar de una forma muy sencilla las distintas funcionalidades que sean añadidas. Dicha ventana tendrá sera como la de la figura D.5



Figura D.5: Ventana de la aplicación compilada y ejecutada.

## Crear una nueva release

Tras realizar las pertinentes modificaciones o añadir nuevas funcionalidades a la aplicación, el siguiente paso será compartir la aplicación. MATLAB para ello tiene tres opciones:

- **MATLAB App:** esta opción permite crear un archivo de instalación para compartir la aplicación con otros usuarios de MATLAB. Si el objetivo es que estos puedan modificar o ver el funcionamiento del código, esta es la mejor opción.
- **Web App:** esta opción permite crear y desplegar aplicaciones web utilizando MATLAB Compiler, facilitando su distribución y uso en diferentes entornos. Esto ha sido desestimado ya que esta funcionalidad la ofrece pero es de pago.
- **Standalone Desktop App:** Esta opción permite crear aplicaciones de escritorio independientes usando MATLAB Compiler, que requieren tener MATLAB instalado para su ejecución. Esta es la opción que he utilizado yo y la que voy a desarrollar un poco más.

## MATLAB Standalone Application

Para crear una Standalone Application dentro de MATLAB App Designer tendremos que pulsar el icono de compartir y seguido de esto seleccionar la opción de ‘Standalone Desktop App’ tras la cual nos aparecerá una ventana para añadir la información relevante de la aplicación que se va a crear. Aquí hemos de introducir los siguientes datos los cuales se ven en la figura D.6:

- Nombre de la aplicación.
- Versión de la aplicación.
- Autor (nombre, correo, compañía).
- Descripción (opcional).
- Archivos y directorios necesarios para la correcta ejecución del programa.
- Archivos visibles para el usuario.

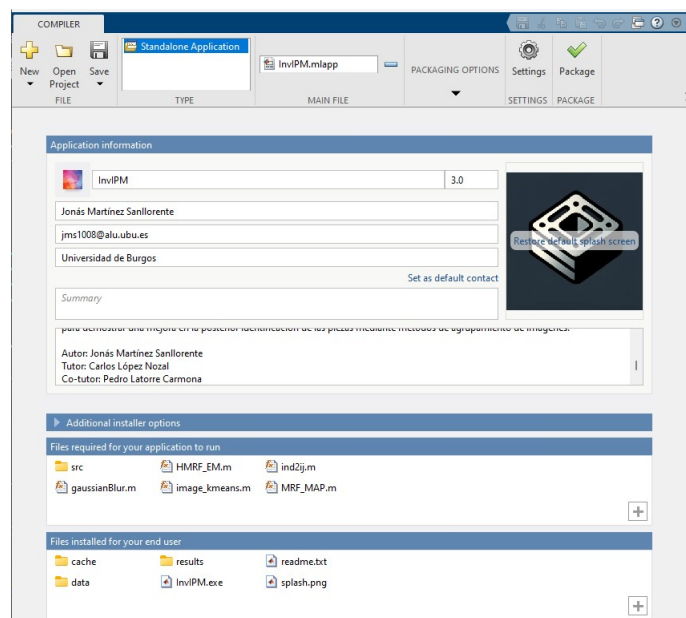


Figura D.6: Creación de MATLAB Standalone App.

## Ejecución de la Standalone App

Al abrir el archivo ‘InvIPM.exe’, se accederá a la aplicación, la cual ejecutará todos los scripts y el código de manera transparente para el usuario, mostrando únicamente los directorios deseados. En mi caso, estos serían:

- El directorio ‘/data’ ya que contienen las imágenes de prueba con sus respectivas imágenes ground truth.
- El directorio ‘/cache’ ya que sera donde se almacenen todos los resultados de antiguas ejecuciones de forma temporal.
- El directorio ‘/results’ ya que sera la ruta por defecto que se le ofrezca al usuario cuando este desee guardar los resultados de una ejecución.
- El README con las pertinentes explicaciones.
- El splash de la aplicación.

## D.5. Pruebas del sistema

### Casos de prueba

Se han creado múltiples tests unitarios para comprobar el correcto funcionamiento de manera aislada de los scripts que utiliza la aplicación.

Los distintos tests creados son los siguientes:

- `alvarez_transformTest.m`
  - Test básico.
  - Test de valores extremos.
  - Test con una imagen uniforme.
  - Test con una imagen en blanco y negro.
- `maddern_transformTest.m`
  - Test básico.
  - Test de valores extremos.
  - Test con una imagen uniforme.



- `krajnik_transformTest.m`
  - Test básico.
  - Test de valores extremos.
  - Test con una imagen uniforme.
- `upcroft_transformTest.m`
  - Test básico.
  - Test de valores extremos.
  - Test con una imagen uniforme.
- `calcular_alphaTest.m`
  - Test valores básicos.
  - Test con  $\lambda_1$  a 0.
  - Test con las tres  $\lambda$ s iguales.
- `calcular_theta_krajnikTest.m`
  - Test valores básicos.
  - Test con una imagen uniforme.
  - Test con una imagen en blanco y negro.
- `calcular_coincidenciaTest.m`
  - Test con imágenes iguales.
  - Test con imágenes diferentes.
- `check_cacheTest.m`
  - Test existe caché (la crea, añade una imagen y luego borra la caché).
- `convertir_a_blanco_negro_con_bordesTest.m`
  - Test con un caso perfecto.
  - Test con un caso no perfecto.
- `imagen_tres_canalesTest.m`
  - Test con una imagen de cuatro canales.

- Test con una imagen de tres canales.
- `metodos_agrupamientoTest.m`
  - Test de K-Means sin imagen ground truth.
  - Test de K-Means con imagen ground truth.
- `metodos_invariantesTest.m`
  - Test de el invariante de Álvarez.
  - Test de el invariante de Maddern.
- `moda_colorTest.m`
  - Test básico, un color aparece repetido.
  - Test de colores únicos.
- `segmentar_imagen_fuzzy_CMeansTest.m`
  - Test básico.
- `segmentar_imagen_GMMTest.m`
  - Test básico.
- `segmentar_imagen_KMeansTest.m`
  - Test básico.

Como se puede apreciar, ni PCA ni HMRF\_EM tienen test unitarios propios ya que dichas librerías ya han sido testeadas por sus desarrolladores.

## Ejecución de los casos de prueba

Para ejecutar los casos de prueba, como viene indicado en el README de la carpeta test, hemos de copiar los scripts del directorio ‘/test’ al directorio ‘/functions’.

Una vez realizado esto tenemos dos opciones.

1. **Ejecutar uno a uno:** si deseas ejecutar un test específico por separado, abre la consola de comandos y utiliza el siguiente comando, reemplazando ‘nombreDelTest’ por el nombre del test que deseas ejecutar ‘`disp(runtests(‘nombreDelTest’))`’.

2. **Ejecutar todos a la vez:** si lo que quieres es ejecutar todos a la vez, simplemente tienes que ejecutar el archivo 'run\_all\_tests.m'. Este archivo se encargará de correr todos los tests de forma automática.

Los resultados de ejecutar todos los tests son los que se pueden observar en la figura [D.7](#).

TestName	Passed	Failed
{'alvarez_transformTest' }	4	0
{'calcular_alphaTest' }	3	0
{'calcular_theta_krajnikTest' }	3	0
{'calcular_coincidenciaTest' }	2	0
{'check_cacheTest' }	1	0
{'convertir_a_blanco_negro_con_bordesTest' }	2	0
{'imagen_tres canalesTest' }	2	0
{'moda_colorTest' }	2	0
{'metodos_invariantesTest' }	2	0
{'metodos_agrupamientoTest' }	2	0
{'segmentar_imagen_KMeansTest' }	1	0
{'segmentar_imagen_fuzzy_CMeansTest' }	1	0
{'segmentar_imagen_GMMTest' }	1	0

Figura D.7: Resultados de la ejecución de los tests.



## *Apéndice E*

---

# Documentación de usuario

---

## E.1. Introducción

Este apéndice está destinado a abordar los temas relacionados con la parte del usuario. Aquí se tratarán los requisitos, la instalación de la aplicación y el manual de uso para el usuario.

## E.2. Requisitos de usuarios

Los requisitos necesarios para poder utilizar la aplicación son:

- Disponer de un ordenador, ya que se trata de una aplicación de escritorio.
- MATLAB 2023b.

### **MATLAB 2023b**

El único programa que es necesario tener para la ejecución de este proyecto es MATLAB, mas concretamente la versión 2023b. Para descargarlo tenemos que ir a la página oficial de MATLAB [\[12\]](#). Una vez iniciada la aplicación nos pedirá introducir una cuenta de MATLAB, la Universidad de Burgos ofrece a los alumnos el acceso con su correo institucional a la versión Academic de MATLAB.

## E.3. Instalación

No hay un proceso de instalación adicional aparte de las dependencias mencionadas anteriormente, ya que la aplicación es auto contenida. Esto significa que todos los archivos y bibliotecas necesarios ya están incluidos en la carpeta de la versión. Gracias a esto, la instalación y el uso se simplifican considerablemente, permitiendo que incluso los usuarios sin experiencia en MATLAB tengan una experiencia agradable.

El primer paso sera ir al repositorio de GitHub y descargarse la ultima release, esto se puede hacer tal y como se muestra en la figura E.1 en el menú de la derecha:

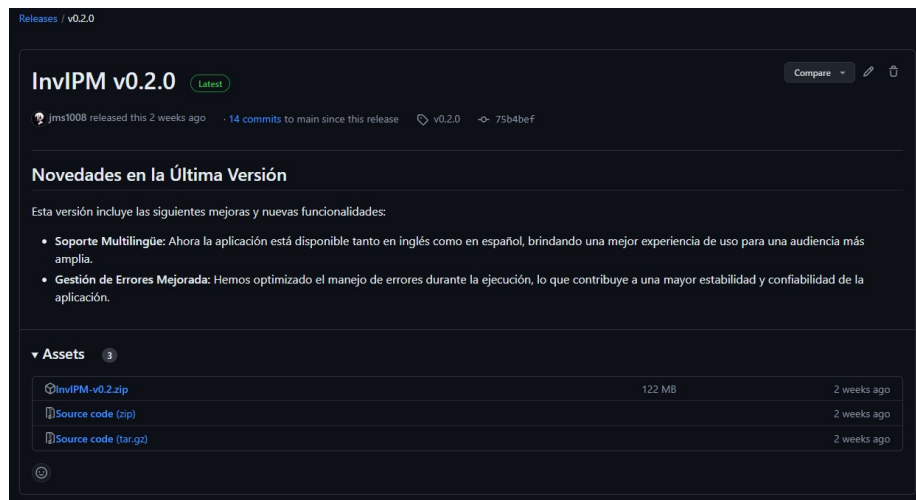


Figura E.1: Instalación de la release desde GitHub.

Una vez descargado el archivo zip tendremos que descomprimirlo. Tras ello, nos quedaran una serie de archivos y carpetas en el directorio deseado. Estas son:

- /InvIPM.exe: este es el ejecutable de la aplicación.
- /readme.txt: es un documento de ayuda el cual nos indica los requisitos necesarios además de como ejecutar la aplicación.
- /data: en este directorio se almacenan las imágenes de ejemplo que trae la propia aplicación, estas podrán ser borradas por el usuario si así lo desea.

- /cache: en este directorio se almacenaran las imágenes guardadas en caché.
- /results: este es el directorio que se le ofrece al usuario por defecto cuando quiere guardar los resultados de una ejecución.
- /splash.png: es la imagen que aparece nada mas abrir la aplicación.

## E.4. Manual del usuario

Sección dedicada a la redacción de un manual sencillo para que el usuario comprenda cómo utilizar la aplicación.

### Bienvenida

Nada mas entrar en la aplicación nos encontraremos con una imagen de bienvenida donde nos explica en términos generales el objetivo de la misma además del autor y tutores del proyecto. El contenido de esta pantalla se muestra en la figura E.2

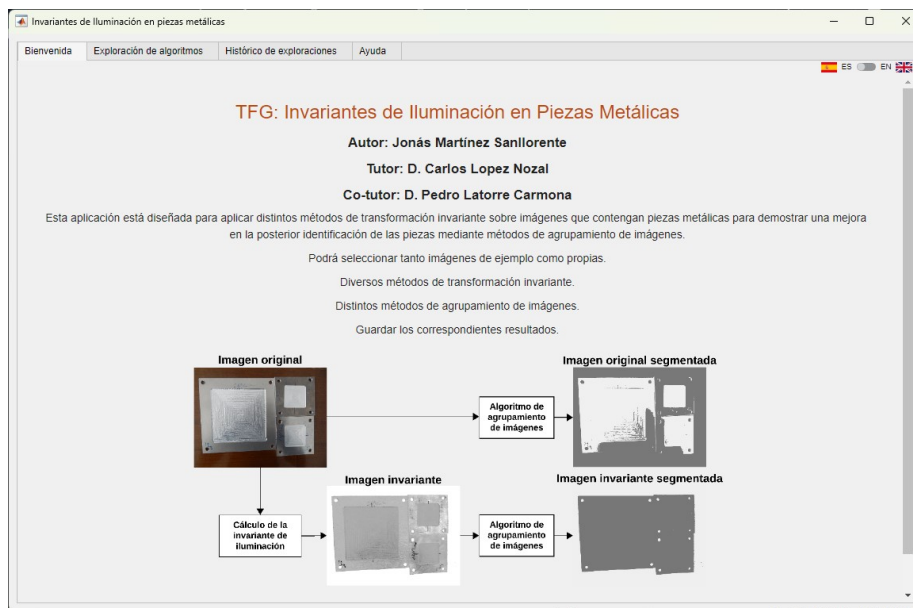


Figura E.2: Apartado de bienvenida de InvIMP.

En la parte superior derecha de esta pantalla, se encuentra un selector de idioma entre español e inglés, con sus correspondientes banderas. Esto lo

hace más autoexplicativo, permitiendo al usuario identificarlo sin necesidad de leer el texto. Esto es especialmente útil cuando la aplicación está en un idioma diferente al del usuario.

## Exploración de algoritmos

En este apartado, es donde el usuario puede cargar, ejecutar y visualizar los resultados de la ejecución.

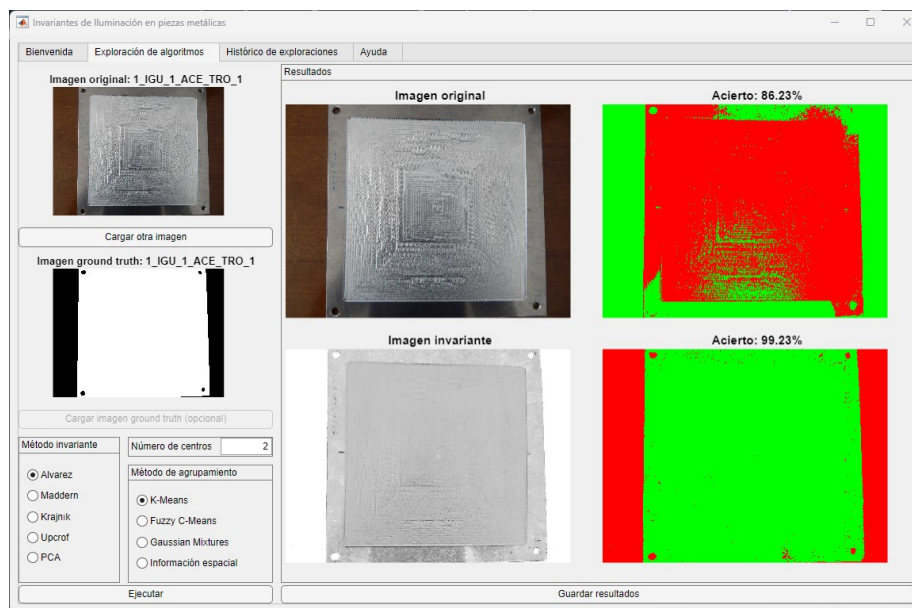


Figura E.3: Apartado de exploración de algoritmos de InvIMP.

### Selección de imagen

Mediante el botón de 'Cargar imagen de pieza metálica' podremos seleccionar una imagen proporcionada por el sistema o una propia sobre la cual aplicar mas adelante.

### Selección de imagen ground truth

En caso de no ser una imagen proporcionada por el sistema se habilitará el botón de 'Cargar imagen ground truth (opcional)' para de esta manera poder proporcionar una propia. Esta imagen como indica el texto no es necesaria para la ejecución exceptuando el obtener el porcentaje de acierto.



### Selección de algoritmo invariante

Se proporcionan los distintos métodos invariantes:

- Álvarez.
- Maddern.
- Krajinik.
- Upcprof.
- PCA.

Estos están ordenados de manera que el primero es el que mejores resultados ofrece según las pruebas realizadas. Por otro lado, PCA, aunque muestra una clara mejoría en muchas situaciones, es un enfoque más clásico que numéricamente no alcanza resultados tan buenos como los demás.

### Selección de algoritmo de agrupamiento

Se proporcionan los distintos métodos de agrupamiento:

- K-Means.
- Fuzzy C-Means.
- Gaussian Mixtures.
- Información espacial, el cual es HMRF\_EM.

Estos están ordenados de manera que el primero es el menos costoso computacionalmente, por lo que ofrecerá mejores tiempos de ejecución. El último, en cambio, tiene tiempos de ejecución más largos, ya que es un algoritmo que no solo considera el color de un píxel en particular, sino también el de los píxeles circundantes.

### Selección de número de centros

Mediante este cuadro de texto el usuario puede indicar el numero de centros a utilizar en los algoritmos de agrupamiento. Por defecto este numero será 2.

El número de centros ha de cumplir las siguientes condiciones:

- Ser un numero positivo.
- Ser igual o mayor que 2.
- Ser igual o menor que 10.

Esto se debe a que, como mínimo, se necesitan dos centros para poder separar entre pieza y fondo. El máximo se establece porque, a mayor cantidad de centros, aumenta la complejidad computacional, lo cual alargaría considerablemente los tiempos de ejecución.

### Ejecución

Al pulsar el botón de ‘Ejecutar’ se dar’a inicio a la ejecución sobre la imagen seleccionada de los distintos algoritmos elegidos. Durante la ejecución, se proporcionará retroalimentación sobre el porcentaje de progreso y la etapa que se está calculando en ese momento.

### Representación de resultados

Tras finalizar la ejecución se mostraran en la parte derecha cuatro imágenes, estas serán:

- La imagen original (arriba a la izquierda).
- La imagen original segmentada (arriba a la derecha).
- La imagen invariante (abajo a la izquierda).
- La imagen invariante segmentada (abajo a la derecha).

En el caso de que se haya utilizado una imagen proporcionada por la aplicación o que el usuario haya proporcionado una imagen de referencia (ground truth), el título tanto de la imagen original segmentada como de la imagen invariante segmentada mostrará el porcentaje de acierto. De esta manera, el usuario podrá comparar los resultados no solo visualmente, sino también numéricamente.

## Histórico de exploraciones

En este apartado, se nos presenta una tabla en la cual aparecen a forma de lista todas aquellas ejecuciones que hemos realizado tanto en esta sesión como en sesiones previas. El contenido de las columnas de esta tabla es el siguiente:

- **Nombre del fichero:** este es el nombre del archivo que el usuario seleccionó.
- **Fecha:** esta es la fecha de la ejecución. Esta está en formato DD-MMM-YYYY HH-MM-SS.
- **Algoritmo invariante:** muestra el nombre del algoritmo invariante aplicado sobre la imagen. En el caso de no haberse aplicado ningún algoritmo invariante muestra 'No aplicado'.
- **Algoritmo de agrupamiento:** muestra el nombre del algoritmo de agrupamiento aplicado sobre la imagen. En el caso de no haberse aplicado ningún algoritmo de agrupamiento muestra 'No aplicado'.
- **Número de centros:** muestra la cantidad de centros utilizados en el algoritmo de agrupamiento aplicado sobre la imagen. En el caso de no haberse aplicado ningún algoritmo de agrupamiento muestra '-'.
- **Imagen ground truth:** indica mediante 'Sí' o 'No' el si la ejecución de dicha imagen tenía o bien imagen asociada o el usuario había proporcionado una propia.
- **Medida de calidad:** indica la tasa de acierto que se ha logrado al comparar la imagen correspondiente a dicha fila con si imagen ground truth. En caso de no tener imagen ground truth esta celda estará en blanco.
- **Archivo:** muestra un checkbox el cual al pulsar sobre el abre en una nueva ventana la imagen correspondiente a dicha fila. Esta nueva ventana tiene una serie de opciones aparte de la visualización permitiendo que el usuario la guarde o imprima entre otras opciones.

El usuario también puede si lo desea ordenar la tabla por la columna que quiera.

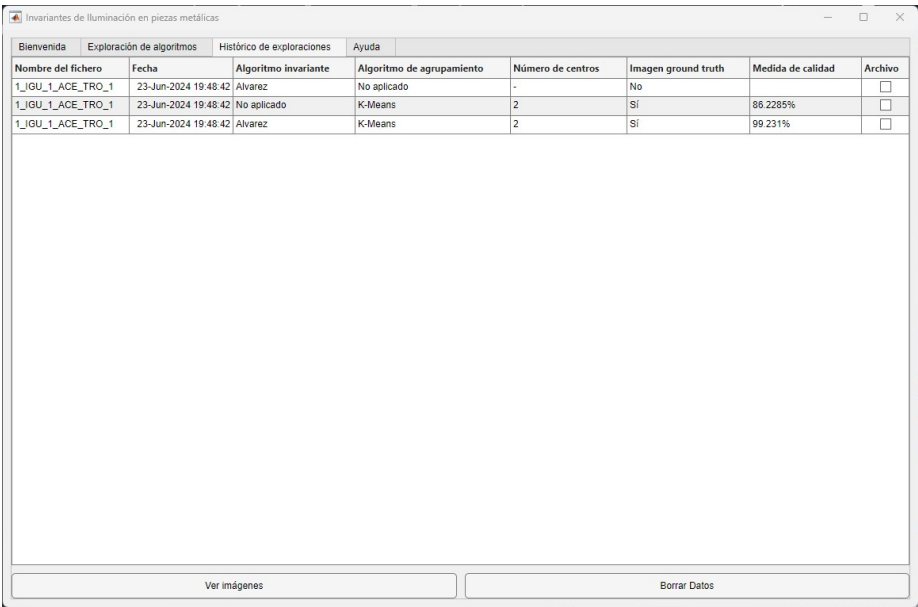


Figura E.4: Apartado de histórico de exploraciones de InvIMP.

A parte de la tabla, en la parte inferior hay dos botones, uno para ver las imágenes y otro para borrar el contenido de la memoria cache.

**Ver imágenes**

Al pulsar en este botón se abrirá una ventana nueva donde podremos ver y seleccionar todas aquellas imágenes que están almacenadas en la memoria caché.

**Borrar datos**

Al pulsar en este botón aparecerá una ventana emergente indicando si realmente queremos borrar de forma irreversible el contenido de la memoria caché. En caso de pulsar en el botón de ‘Borrar’ se borrará su contenido además de actualizarse la tabla ya que dichas imágenes no se encontraran disponibles. Tras esto nos indicará que la memoria cache se ha borrado exitosamente.

**Ayuda**

En este apartado, tal y como se muestra en la figura E.5, se puede obtener información sobre el funcionamiento, los algoritmos invariantes, los

algoritmos de agrupamiento, la caché y la tasa de acierto. Al estar la propia documentación implementada dentro de la aplicación, resulta muy cómodo buscar información sobre cualquier tema de interés.

Además, el idioma de la documentación corresponde con el que se ha seleccionado en la bienvenida.

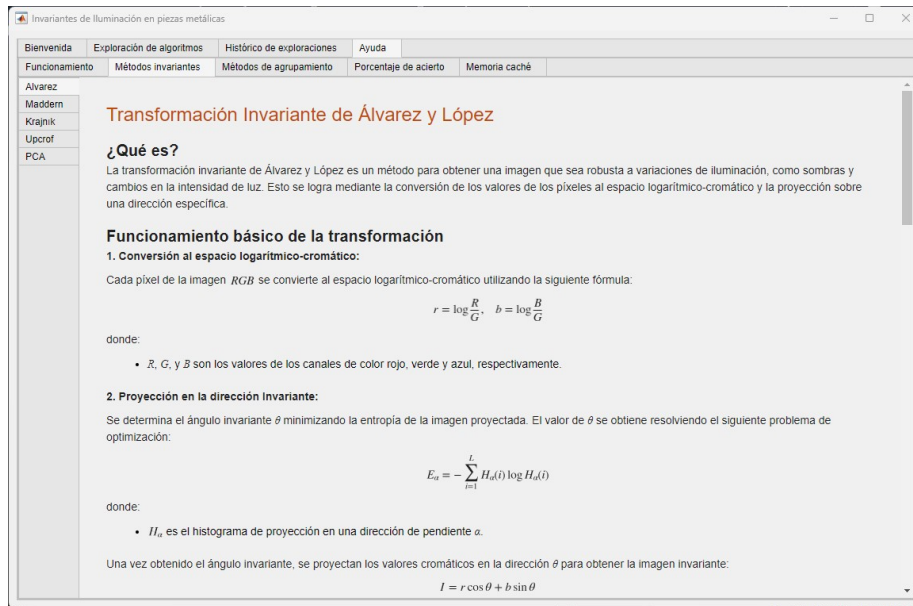


Figura E.5: Apartado de ayuda de InvIMP.



## *Apéndice F*

---

# **Anexo de sostenibilización curricular**

---

### **F.1. Introducción**

Este anexo incluirá una reflexión personal del alumnado sobre los aspectos de la sostenibilidad que se abordan en el trabajo. Se pueden incluir tantas subsecciones como sean necesarias con la intención de explicar las competencias de sostenibilidad adquiridas durante el alumnado y aplicadas al Trabajo de Fin de Grado.

Más información en el documento de la CRUE [https://www.crue.org/wp-content/uploads/2020/02/Directrices\\_Sostenibilidad\\_Crue2012.pdf](https://www.crue.org/wp-content/uploads/2020/02/Directrices_Sostenibilidad_Crue2012.pdf).

Este anexo tendrá una extensión comprendida entre 600 y 800 palabras.





---

## Bibliografía

---

- [1] CC BY-NC-SA 4.0 Deed | Atribución-NoComercial-CompartirIgual 4.0 Internacional | Creative Commons — [creativecommons.org](https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es). <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>. [Accessed 17-06-2024].
- [2] Ieee recommended practice for software requirements specifications. *IEEE Std 830-1998*, pages 1–40, 1998.
- [3] José M Álvarez Alvarez and Antonio M. López. Road detection based on illuminant invariance. *IEEE Transactions on Intelligent Transportation Systems*, 12(1):184–193, 2011.
- [4] GitHub. Github desktop download. [https://desktop.github.com/?ref\\_cta=download+desktop&ref\\_loc=installing+github+desktop&ref\\_page=docs](https://desktop.github.com/?ref_cta=download+desktop&ref_loc=installing+github+desktop&ref_page=docs). [Accessed 21-06-2024].
- [5] Glassdoor. Sueldos para el puesto de Ingeniero Informático en España. [https://www.glassdoor.es/Sueldos/ingeniero-inform%C3%A1tico-sueldo-SRCH\\_K00,21.htm](https://www.glassdoor.es/Sueldos/ingeniero-inform%C3%A1tico-sueldo-SRCH_K00,21.htm). [Accessed 16-06-2024].
- [6] Taeyoung Kim, Yu-Wing Tai, and Sung-Eui Yoon. Pca based computation of illumination-invariant space for road detection. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 632–640, 2017.
- [7] Tomáš Krajník, Jan Blažíček, and João M. Santos. Visual road following using intrinsic images. In *2015 European Conference on Mobile Robots (ECMR)*, pages 1–6, 2015.

- [8] Will Maddern, Alex Stewart, Colin McManus, Ben Upcroft, Winston Churchill, and Paul Newman. Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles. In *Proceedings of the Visual Place Recognition in Changing Environments Workshop, IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China*, volume 2, page 5, 2014.
- [9] MathWorks Inc. *Fuzzy Logic Toolbox: Design and Simulate Fuzzy Logic Systems*, 2023. Version 10.3 (R2023b).
- [10] MathWorks Inc. *MATLAB: The Language of Technical Computing*, 2023. Version 10.3 (R2023b).
- [11] MathWorks Inc. *Statistics and Machine Learning Toolbox: Use fitgmdist for Fitting Gaussian Mixture Models*, 2023. Version 10.3 (R2023b).
- [12] MATLAB. Obtener productos de matlab y simulink. <https://es.mathworks.com/downloads/>. [Accessed 21-06-2024].
- [13] Ben Upcroft, Colin McManus, Winston Churchill, Will Maddern, and Paul Newman. Lighting invariant urban street classification. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1712–1718, 2014.
- [14] Quan Wang. Hmrf-em-image: implementation of the hidden markov random field model and its expectation-maximization algorithm. *arXiv preprint arXiv:1207.3510*, 2012.