



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 — 17 de junio
de 2024

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



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Carlos López Nozal y D. Pedro Latorre Carmona, profesores del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Exponen:

Que el alumno D. Jonás Martínez Sanllorente, con DNI 71363040C, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado “Aplicación de transformaciones invariantes a la iluminación para mejora de la segmentación en imágenes. Aplicación a piezas metálicas”.

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

En Burgos, 17 de junio de 2024

Vº. Bº. del Tutor:

D. Carlos López Nozal

Vº. Bº. del co-tutor:

D. Pedro Latorre Carmona

Resumen

El proceso de análisis visual piezas metálicas en imágenes se ve influido por diversos factores. Algunos de los más importantes están directamente relacionados con las condiciones de iluminación de la escena. En concreto, la forma como se adquieren las imágenes de las propias piezas metálicas. Dado que los materiales metálicos son altamente reflectantes, se producen tanto sombras como reflexiones de tipo especular [1] que degradan sustancialmente la propia calidad visual de la imagen.

En el caso se quisieran aplicar métodos de segmentación de imágenes sobre ellas, se producirían errores en la generación de las diferentes regiones a identificar en las mismas. Es, por tanto, necesario aplicar algún tipo de pre-procesamiento que mitigue dichas fuentes de error, en la medida de lo posible.

En este trabajo se propone analizar el rendimiento de una serie de técnicas de transformación sobre las imágenes en color adquiridas denominadas *transformaciones invariantes de la iluminación*, aplicadas en forma de técnica(s) de pre-procesamiento. Por otro lado, se creará una aplicación / aplicación web para que el usuario pueda aplicar dichas transformaciones, así como métodos de segmentación (agrupamiento o *clustering*) de una forma intuitiva y rápida.

Descriptores

Piezas metálicas, reflexión metálica, identificación de piezas, métodos invariantes, segmentación de imágenes, aplicación.

Abstract

Reflections on metal parts and the shadows that can be caused are a very common problem when it comes to correctly identifying the limits of a part.

In most cases, this means that after applying segmentation methods, erroneous identification of the parts is produced due to the fact that obtaining images in ideal conditions where neither reflections nor shadows are found is very complicated and trying to replicate a scenario of this kind can be very costly.

This is why by applying pre-processing methods to the images taken of the pieces, it is possible to obtain a result that has these reflections, shadows and other image defects mitigated so that when the image is subsequently segmented, the result will be of a higher quality.

This work proposes the demonstration of a better result when segmenting after having applied invariant methods to the initial image.

In addition, an application so that the user, can apply the different algorithms in an intuitive, fast and comfortable way.

Keywords

Metallic parts, metallic reflection, part identification, invariant methods, image segmentation, application

Índice general

Índice general	iii
Índice de figuras	v
Índice de tablas	vi
1. Introducción	1
1.1. Estructura de la memoria	2
1.2. Materiales adjuntos	3
2. Objetivos del proyecto	5
2.1. Objetivos generales	5
2.2. Objetivos técnicos	5
2.3. Objetivos personales	6
3. Conceptos teóricos	7
3.1. Preprocesado de imágenes	7
3.2. Métodos de transformación invariante	8
3.3. Métodos de agrupamiento de imágenes	17
3.4. Clasificación de resultados	26
4. Técnicas y herramientas	29
5. Aspectos relevantes del desarrollo del proyecto	31
6. Trabajos relacionados	33
7. Conclusiones y Líneas de trabajo futuras	35

Índice de figuras

3.1. Resultado de aplicar Álvarez	9
3.2. Resultado de aplicar Maddern	11
3.3. Resultado de aplicar Krajník	13
3.4. Resultado de aplicar Upcroft	15
3.5. Resultado de aplicar PCA	16
3.6. Resultado de aplicar K-Means	19
3.7. Resultado de aplicar Fuzzy C-Means	21
3.8. Resultado de aplicar Gaussian Mixtures Models	23
3.9. Resultado de aplicar Hidden Markov Random Field	25
3.10. Resultado de crear la imagen “ground truth”	26

Índice de tablas

1. Introducción

En el ámbito de la fabricación, las piezas metálicas son fundamentales para una amplia variedad de aplicaciones industriales. La precisión en su fabricación es crucial para garantizar la funcionalidad y durabilidad de los productos finales.

El control de calidad en el proceso de producción es de gran importancia, ya que asegura que los productos finales cumplan con los estándares establecidos y satisfagan las expectativas del cliente. A través de técnicas de inspección y pruebas rigurosas, se pueden identificar y corregir defectos en etapas tempranas del proceso de producción. Esto no solo reduce el riesgo de fallos en el producto terminado, sino que también optimiza el uso de materiales y recursos, incrementando la eficiencia general.

Comúnmente, esta inspección técnica y pruebas suelen ser llevadas a cabo por un operario, ya que los sistemas basados en la toma de imágenes para detectar fallos en el producto dependen en gran medida de la calidad de la fotografía tomada. En la mayoría de las ocasiones, a menos que el fondo sea muy fácilmente distinguible de la pieza y las condiciones de iluminación sean ideales, los métodos de agrupamiento difícilmente pueden identificar la pieza correctamente. Por esta razón, los métodos de transformación invariante desempeñan un papel esencial, ya que permiten compensar y corregir las distorsiones causadas por variaciones en las condiciones de iluminación, como reflejos y sombras los cuales en piezas metálicas se dan con mucha facilidad.

Los métodos invariantes son responsables de generar una nueva imagen a partir de la original en la cual las reflexiones de luz, las sombras o errores inducidos por las condiciones lumínicas en el momento de la captura de imágenes estén mitigados. Al aplicar sobre esta nueva imagen algún método

de agrupamiento de imágenes, se obtendrá un resultado mejor, o dicho en otras palabras, más cercano a la realidad.

Un método de agrupamiento de imágenes sirve para segmentarlas en diferentes partes, identificando y clasificando áreas similares dentro de la imagen. Este proceso es crucial para separar la pieza metálica del fondo y otros elementos no deseados, facilitando así una inspección técnica más precisa.

La implementación de estos métodos de manera previa a la segmentación de imágenes no solo mejora la precisión, sino que también reduce la variabilidad que se da por diferencias en la iluminación del entorno. Esto es particularmente importante en entornos de producción industrial, donde las condiciones de iluminación pueden no ser siempre ideales y variar significativamente, afectando la visibilidad y el contraste en las imágenes capturadas.

En resumen, el uso de métodos de transformación invariantes en la segmentación de imágenes de piezas metálicas tiene un papel muy importante a la hora de contrarrestar los efectos adversos de las condiciones de iluminación. Estas técnicas no solo mejoran considerablemente la precisión de los resultados y la consistencia en la correcta identificación de las piezas metálicas en imágenes, sino que permiten automatizar el control de calidad dentro de la propia industria, optimizando así los procesos de producción.

1.1. Estructura de la memoria

La memoria sigue la siguiente estructura:

- **Introducción:** breve descripción del problema a resolver y la solución propuesta. Estructura de la memoria y listado de materiales adjuntos.
- **Objetivos del proyecto:** exposición de los objetivos que persigue el proyecto.
- **Conceptos teóricos:** breve explicación de los conceptos teóricos clave para la comprensión de la solución propuesta.
- **Técnicas y herramientas:** listado de técnicas metodológicas y herramientas utilizadas para gestión y desarrollo del proyecto.
- **Aspectos relevantes del desarrollo:** exposición de aspectos destacables que tuvieron lugar durante la realización del proyecto.
- **Trabajos relacionados:** estado del arte en el campo de la monitorización de la actividad de vuelo de colmenas y proyectos relacionados.

- **Conclusiones y líneas de trabajo futuras:** conclusiones obtenidas tras la realización del proyecto y posibilidades de mejora o expansión de la solución aportada.

Junto a la memoria se proporcionan los siguientes anexos:

- **Plan del proyecto software:** planificación temporal y estudio de viabilidad del proyecto.
- **Especificación de requisitos del software:** se describe la fase de análisis; los objetivos generales, el catálogo de requisitos del sistema y la especificación de requisitos funcionales y no funcionales.
- **Especificación de diseño:** se describe la fase de diseño; el ámbito del software, el diseño de datos, el diseño procedural y el diseño arquitectónico.
- **Manual del programador:** recoge los aspectos más relevantes relacionados con el código fuente (estructura, compilación, instalación, ejecución, pruebas, etc.).
- **Manual de usuario:** guía de usuario para el correcto manejo de la aplicación.

1.2. Materiales adjuntos

Los materiales que se adjuntan con la memoria son:

- Aplicación de escritorio para probar tanto con imágenes propias como con algunas de ejemplo.
- Proyecto de MATLAB para poder probar tanto con imágenes propias como con algunas de ejemplo además de tener acceso al código para añadir o modificar los algoritmos que se utilizan.

Además, los siguientes recursos están accesibles a través de internet:

- Repositorio del proyecto. [GitHub](#)

2. Objetivos del proyecto

2.1. 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 [2], Maddern [5], Krajník [4], Upcroft [9] y PCA [3]) como de segmentación de imágenes (K-Means [7], Fuzzy C-Means [6], GMM [8] y segmentación con información espacial [10]).
- 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.

2.2. Objetivos técnicos

- Comparar los resultados de distintos métodos de transformación invariante de imágenes y argumentar la mejora que se obtiene en la identificación de piezas metálicas.
- Utilizar la plataforma GitHub como sistema de control de versiones
- Utilizar Zube como herramienta de gestión de proyectos.
- Realizar tests unitarios, de integración y de interfaz..
- Utilizar un sistema de documentación.

2.3. Objetivos personales

- Realizar una aportación a la modernización de la industria.
- Profundizar en el uso de MATLAB y el cual tiene una gran potencia computacional, destacando en cálculos matriciales.
- Explorar métodos de visión artificial.
- La creación de sistemas que permitan la automatización y mejoren el control de calidad en diversos entornos.

3. Conceptos teóricos

3.1. Preprocesado de imágenes

Eliminación del canal alpha

El canal alpha, que representa la transparencia de los píxeles, no es relevante ya que solo nos interesa la información de color. Al eliminar este canal reducimos la complejidad de los datos, en nuestro caso imágenes para centrarnos únicamente en los tres canales RGB (rojo, verde y azul)

Para eliminar el canal alpha de una imagen con componentes de color R , G , B y α , tan solo hay que guardar en una nueva imagen lo siguiente:

$$\text{imagen_sin_alpha}(i, j) = \begin{bmatrix} R(i, j) \\ G(i, j) \\ B(i, j) \end{bmatrix}$$

donde:

- $R(i, j)$ es el valor del canal rojo en el píxel (i, j)
- $G(i, j)$ es el valor del canal verde en el píxel (i, j)
- $B(i, j)$ es el valor del canal azul en el píxel (i, j)

Eliminamos el canal alpha $\alpha(i, j)$, que representa la transparencia, ya que no es necesario en este caso.

3.2. Métodos de transformación invariante

Los métodos de transformación invariante son distintos algoritmos utilizados para mitigar o incluso eliminar las variaciones en la iluminación en una imagen. Aunque comúnmente suelen ser usados para obtener resultados similares en un mismo escenario pero variando la cantidad de luz ambiental o la cantidad de focos de luz, también son muy útiles para mitigar los efectos de las sombras y reflejos que comúnmente se ocasionan al fotografiar piezas metálicas.

A continuación se presentan cinco métodos distintos de transformación invariante para poder comparar los resultados.

Álvarez

La transformación invariante de Álvarez y López [2] es un método para obtener una imagen que sea robusta a variaciones de iluminación, como sombras y cambios en la intensidad de luz. Esto se logra mediante la conversión de los valores de los píxeles al espacio logarítmico-cromático y la proyección sobre una dirección específica.

El funcionamiento básico de la transformación invariante de Álvarez y López es el siguiente:

1. Conversión al espacio logarítmico-cromático

Cada píxel de la imagen *RGB* se convierte al espacio logarítmico-cromático utilizando la siguiente fórmula:

$$r = \log \frac{R}{G}, \quad b = \log \frac{B}{G}$$

donde:

- R , G , y B son los valores de los canales de color rojo, verde y azul, respectivamente.

2. Proyección en la dirección Invariante

Se determina el ángulo invariante θ minimizando la entropía de la imagen proyectada. El valor de θ se obtiene resolviendo el siguiente problema de optimización:

$$E_\alpha = - \sum_{i=1}^L H_\alpha(i) \log H_\alpha(i)$$

donde:

- H_α es el histograma de proyección en una dirección de pendiente α .

Una vez obtenido el ángulo invariante, se proyectan los valores cromáticos en la dirección θ para obtener la imagen invariante:

$$I = r \cos \theta + b \sin \theta$$

En la figura 3.1 se puede ver el resultado de aplicar este método invariante.

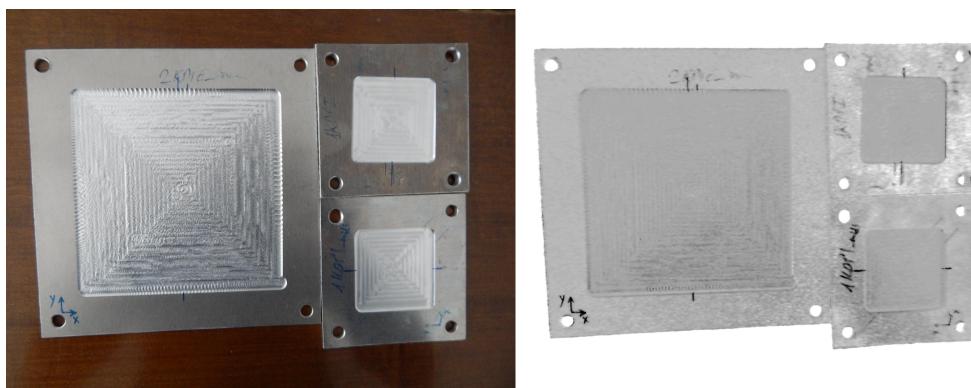


Figura 3.1: Resultado de aplicar Álvarez

Maddern

La transformación invariante de Maddern [5] es un método que genera imágenes invariantes a cambios en la iluminación, reduciendo la influencia de sombras y variaciones en la intensidad de la luz. Se basa en un modelo de espectro del sensor de la cámara y la luz incidente.

El funcionamiento básico de la transformación invariante de Maddern es el siguiente:

1. Relación entre canales de color

Dada una cámara con sensibilidad espectral $F(\lambda)$, el modelo de respuesta de un canal específico R a la luz de una longitud de onda particular λ se define como:

$$R^{x,E} = a^x \cdot n^x \cdot I^x \int S^x(\lambda) E(\lambda) F(\lambda) d\lambda$$

donde:

- a^x y n^x : vectores que representan la dirección de la luz y la normal de la superficie.
- I^x : intensidad de la luz en el punto x .
- $S^x(\lambda)$: reflectancia espectral de la superficie.
- $E(\lambda)$: espectro de la fuente de luz.

2. Espacio logarítmico-cromático

Se transforman las relaciones de los canales de color al espacio logarítmico-cromático tomando el logaritmo de la respuesta espectral:

$$\log(R^{x,E}) = \log(G^x I^x) + \log S^x(\lambda_i) + \log E^x(\lambda_i)$$

donde:

- $G^x = a^x \cdot n^x$ representa la geometría entre la luz y la escena.

3. Proyección invariante

Se proyectan las respuestas espetrales en una dirección específica que elimina el componente de iluminación. La proyección final en la dirección α se realiza utilizando:

$$I = \log(R_2) - \alpha \log(R_1) - (1 - \alpha) \log(R_3)$$

donde:

- R_1, R_2, R_3 : respuestas espetrales de tres canales de color (e.g. R, G, B).

- α : parámetro determinado por las sensibilidades espectrales de los canales.

$$I = \log(R_2) - \alpha \log(R_1) - (1 - \alpha) \log(R_3)$$

4. Cálculo del parámetro α

El parámetro α se calcula para garantizar que la proyección sea independiente de la temperatura de color T de la luz:

$$\frac{1}{\lambda_2} = \frac{\alpha}{\lambda_1} + \frac{1 - \alpha}{\lambda_3}$$

5. Transformación invariante de Maddern

Finalmente, se aplica la siguiente transformación para obtener una imagen invariante:

$$I_{Maddern} = 0,5 + \log(G) - \alpha \log(B) - (1 - \alpha) \log(R)$$

En la figura 3.2 se puede ver el resultado de aplicar este método invariante.

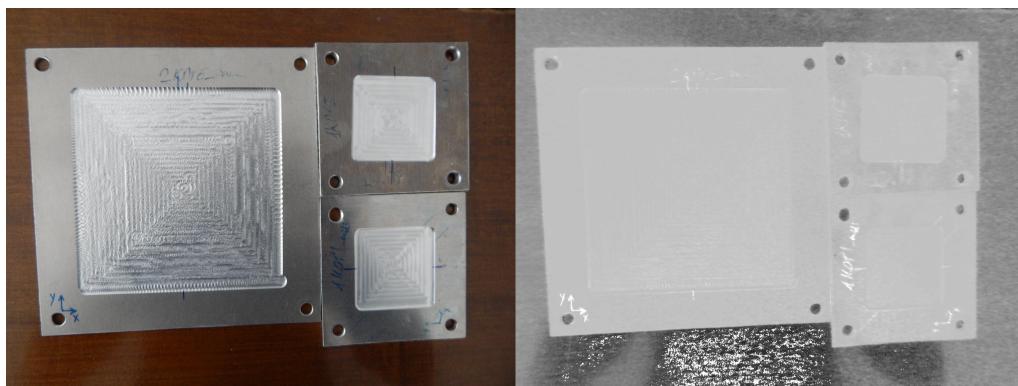


Figura 3.2: Resultado de aplicar Maddern

Krafník

La transformación invariante de Krafník [4] es un método para obtener imágenes invariantes a cambios de iluminación. Se basa en el uso de imágenes intrínsecas, que separan las propiedades intrínsecas de la superficie (reflectancia) de las propiedades extrínsecas (iluminación). Este enfoque permite reducir la influencia de sombras y variaciones de iluminación en la imagen.

El funcionamiento básico de la transformación invariante de Krafník es el siguiente:

1. Modelo Espectral *RGB*

La respuesta espectral de un canal específico ρ_k (con $k = \{r, g, b\}$) se define como:

$$\rho_k = \sigma \int E(\lambda) S(\lambda) Q_k(\lambda) d\lambda$$

donde:

- σ : factor constante que denota el sombreado de Lambert.
- $E(\lambda)$: iluminación incidente en el elemento de superficie percibida.
- $S(\lambda)$: reflectancia espectral de la superficie.
- $Q(\lambda)$: sensibilidad espectral del canal k .
- λ : longitud de onda.

2. Transformación Logarítmica

Se transforma el modelo a un espacio logarítmico para hacer que sea invariante a la iluminación. Esto se realiza calculando el logaritmo de la respuesta espectral relativa a un canal de referencia (e.g., azul):

$$\chi_{k,g} = \log \frac{\rho_{r,g}}{\rho_b}$$

3. Proyección a un Espacio Invariante

Se proyecta el espacio logarítmico-cromático en una dirección θ para obtener la imagen invariante.

$$I_\theta = \chi_r \cos \theta + \chi_g \sin \theta$$

4. Minimización de la Entropía

Se determina el ángulo θ que minimiza la entropía de la imagen proyectada, para que sea lo más invariante posible a la iluminación.

$$E_\theta = - \sum_i H_\theta(i) \log H_\theta(i)$$

donde:

- $H_\theta(i)$: histograma proyectado en una dirección θ .

En la figura 3.3 se puede ver el resultado de aplicar este método invariante

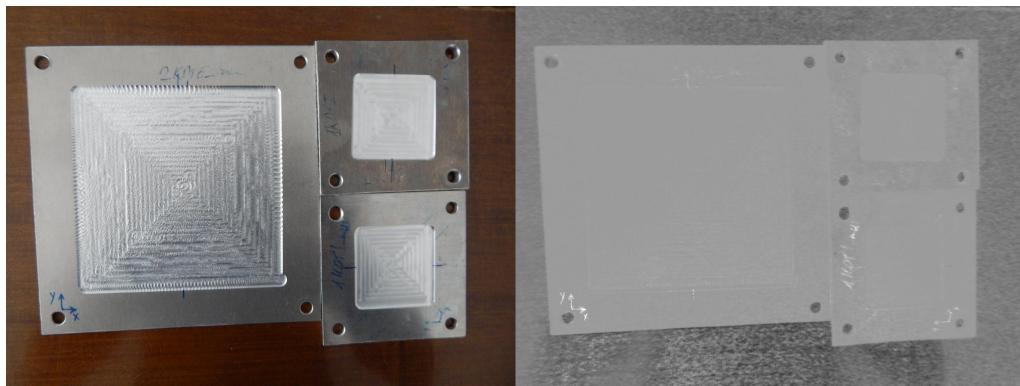


Figura 3.3: Resultado de aplicar Krajník

Upcroft

La transformación invariante de Upcroft [9] es un método que genera imágenes robustas a variaciones en la iluminación, incluyendo sombras y cambios de intensidad. El método se basa en la relación logarítmica entre los canales de color y se utiliza para mejorar la clasificación de escenas urbanas en entornos cambiantes.

El funcionamiento básico de la transformación invariante de Upcroft es el siguiente:

1. Modelo espectral *RGB*

Aunque no aparece explícitamente en el artículo, se menciona que el método de transformación invariante se basa en la relación logarítmica entre los canales *RGB* y hace referencia a trabajos anteriores como los de Ratnasingam y Collins. La respuesta espectral de un canal específico ρ_k (con $k = \{r, g, b\}$) se define como:

$$\rho_k = \sigma \int E(\lambda) S(\lambda) Q_k(\lambda) d\lambda$$

donde:

- σ : factor constante que denota el sombreado de Lambert.
- $E(\lambda)$: iluminación incidente en el elemento de superficie percibida.
- $S(\lambda)$: reflectancia espectral de la superficie.
- $Q(\lambda)$: sensibilidad espectral del canal k .
- λ : longitud de onda.

2. Transformación logarítmica

La transformación invariante utiliza la relación logarítmica entre los canales *RGB* para minimizar la variabilidad causada por la geometría, la intensidad y el espectro de la fuente de luz.

$$F = \log(G) - \alpha \log(B) - (1 - \alpha) \log(R)$$

3. Cálculo del parámetro α

El parámetro α se calcula para minimizar la variabilidad de la fuente de luz. Esto se logra utilizando las longitudes de onda pico de los sensores:

$$\frac{1}{\lambda_G} = \frac{\alpha}{\lambda_B} + \frac{1 - \alpha}{\lambda_R}$$

En la figura 3.4 se puede ver el resultado de aplicar este método invariante.

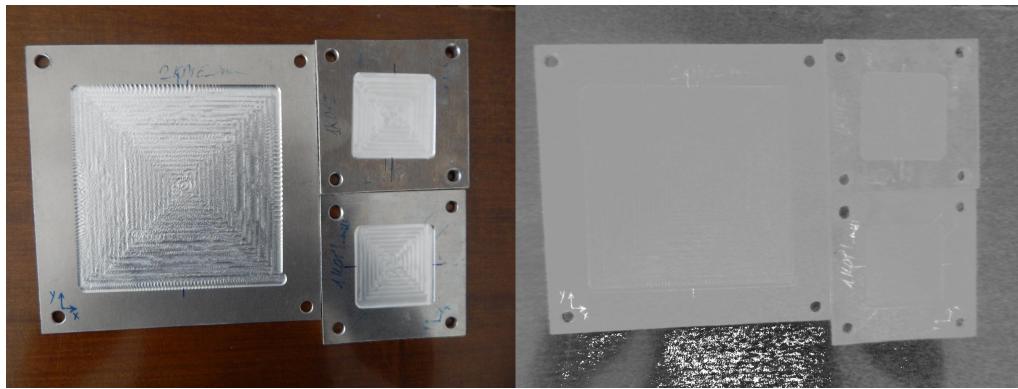


Figura 3.4: Resultado de aplicar Upcroft PCA

PCA

El método basado en PCA (Análisis de Componentes Principales) [3] propuesto por Kim genera un espacio invariante a la iluminación que permite reducir el efecto de sombras y variaciones en la intensidad de la luz. El algoritmo busca la proyección óptima en el espacio logarítmico-cromático para minimizar la variabilidad causada por la iluminación.

El funcionamiento básico de la transformación invariante PCA es el siguiente:

1. Modelo espectral *RGB*

La respuesta espectral de un canal específico ρ_k (con $k = \{r, g, b\}$) se define como:

$$\rho_k = \sigma \int E(\lambda) S(\lambda) Q_k(\lambda) d\lambda$$

donde:

- σ : factor constante que denota el sombreado de Lambert.
- $E(\lambda)$: iluminación incidente en el elemento de superficie percibida.
- $S(\lambda)$: reflectancia espectral de la superficie.
- $Q(\lambda)$: sensibilidad espectral del canal k .
- λ : longitud de onda.

2. Transformación logarítmica

El algoritmo convierte el modelo a un espacio logarítmico para minimizar el efecto de la iluminación. Se calculan las siguientes relaciones logarítmicas:

$$r = \log \frac{R}{(RGB)^{1/3}}, \quad b = \log \frac{B}{(RGB)^{1/3}}$$

donde:

- $(RGB)^{1/3}$: media geométrica de los tres canales.

3. Proyección basada en PCA

Se proyectan los valores en un espacio logarítmico-cromático, donde se encuentra la dirección óptima utilizando PCA:

$$I_\theta = r \cos \theta + b \sin \theta$$

Esto se logra aplicando el Análisis de Componentes Principales a los valores calculados.

En la figura 3.5 se puede ver el resultado de aplicar este método invariante.

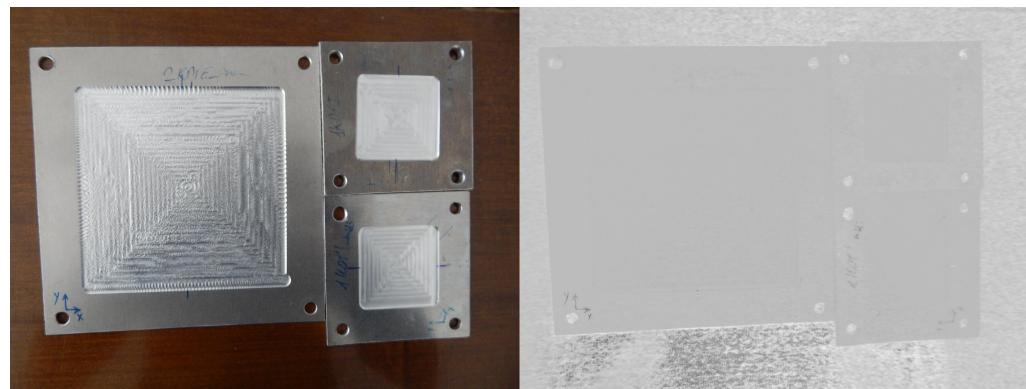


Figura 3.5: Resultado de aplicar PCA

3.3. Métodos de agrupamiento de imágenes

Los métodos de agrupamiento son técnicas de aprendizaje no supervisado que se utilizan para dividir un conjunto de datos en grupos o clústeres, de manera que los datos en el mismo clúster tengan una mayor similitud entre sí que con los datos de otros clústeres. Estos métodos los utilizaremos para separar la pieza metálica del fondo y ver realmente si una transformación invariante previa supone algún cambio significativo.

K-Means

K-Means [7] es un algoritmo de clustering (agrupamiento) utilizado en análisis de datos y aprendizaje automático para dividir un conjunto de datos en un número específico de grupos (o clusters). El objetivo del algoritmo es asignar cada valor de los datos a un grupo de tal manera que la suma de las distancias cuadradas entre los valores de los datos y el centroide de su grupo (el punto medio de todos los valores en ese grupo) sea minimizada. Cada uno de estos grupos es definido por su centroide, que inicialmente es seleccionado al azar y se ajusta iterativamente basándose en los datos asignados al grupo.

Es especialmente útil en aplicaciones de procesamiento de imágenes para la segmentación de imágenes, donde se pueden identificar diferentes regiones basadas en las características de los píxeles. Este método es popular debido a su simplicidad y eficiencia.

Destaca en la segmentación de imágenes, permitiendo dividir la imagen en partes que representan áreas de interés diferentes, facilitando el análisis o procesamiento posterior de esas áreas específicas.

El funcionamiento básico de K-Means es el siguiente:

1. Inicialización

Se seleccionan aleatoriamente k centroides.

2. Asignación

Cada pixel en el conjunto de datos x_i es asignado al centroide más cercano utilizando la siguiente fórmula

$$C_i = \arg \min_{j \in \{1, \dots, k\}} \|x_i - \mu_j\|^2$$

donde:

- C_i es el cluster asignado al punto x_i ,
- x_i es el pixel en la imagen,
- μ_j es el centroide del cluster j ,
- k es el número de clusters.

3. Actualización de Centroides

Los centroides se recalculan como el promedio de los valores de todos los píxeles asignados a su cluster.

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$$

donde:

- μ_j es el nuevo centroide del cluster j ,
- C_j es el conjunto de píxeles asignados al cluster j .

4. Repetición

Los pasos 2 y 3 se repiten hasta que los centroides no cambian significativamente entre iteraciones, lo que indica que se ha alcanzado la convergencia. Siendo el umbral de convergencia por defecto $\epsilon = 1e - 4$.

En la figura 3.6 se puede ver el resultado de aplicar este método de agrupamiento.

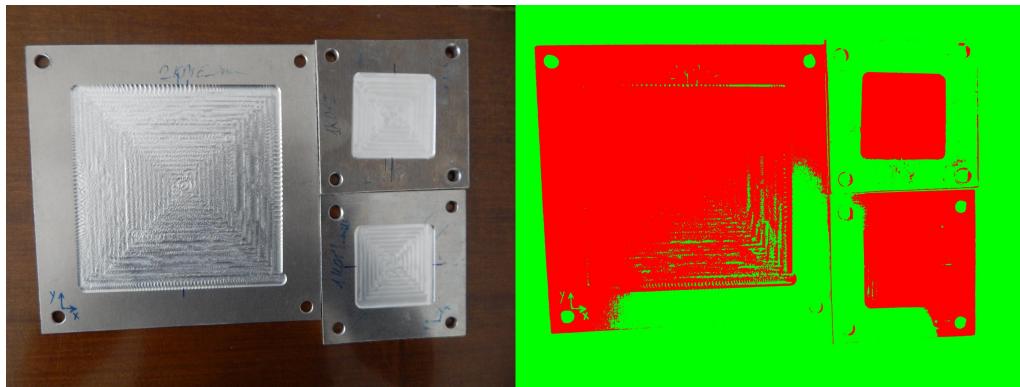


Figura 3.6: Resultado de aplicar K-Means

Fuzzy C-Means

Fuzzy C-Means (FCM) [6] es un algoritmo de clustering basado en la lógica difusa que permite a un punto de datos pertenecer a más de un grupo (o cluster) con diferentes grados de pertenencia. A diferencia de K-Means, donde cada punto pertenece exclusivamente a un solo cluster, en FCM cada punto tiene un valor de pertenencia para cada cluster, lo que indica el grado con el que un punto pertenece a un cluster específico.

FCM es especialmente útil en aplicaciones donde los límites entre las clases no están claramente definidos, lo que ocurre comúnmente en problemas de segmentación de imágenes. De esta forma, al segmentar una imagen que tenga regiones superpuestas con diferentes grados de pertenencia dará un resultado mejor, lo que es útil para segmentar áreas con transiciones suaves.

El funcionamiento básico de Fuzzy C-Means es el siguiente:

1. Inicialización

Se establece el número de clusters c , el parámetro de difusividad m (usualmente 2), y se inicializa la matriz de pertenencias aleatoriamente, denotando la probabilidad de que cada punto pertenezca a cada cluster.

2. Cálculo de centroides

Los centroides se calculan utilizando los valores de pertenencia. El centroide de cada cluster se calcula como:

$$v_j = \frac{\sum_{i=1}^n (u_{ij})^m \cdot x_i}{\sum_{i=1}^n (u_{ij})^m}$$

donde:

- u_j es el centroide del cluster j ,
- u_{ij} es el valor de pertenencia del pixel x_i al cluster j ,
- m es el parámetro de difusividad,
- n es el número total de píxeles.

3. Actualización de valores de pertenencia

Se actualizan los valores de pertenencia para cada pixel en cada cluster utilizando la siguiente fórmula:

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - v_k\|}{\|x_i - v_j\|} \right)^{\frac{2}{m-1}}}$$

donde:

- u_{ij} es el valor de pertenencia del pixel x_i al cluster j ,
- v_j es el centroide del cluster j ,
- m es el parámetro de difusividad,
- c es el número total de clusters.

4. Repetición

Se repiten los pasos 2 y 3 hasta que la diferencia entre las actualizaciones de la matriz de pertenencias en iteraciones sucesivas sea menor que un umbral determinado.

Hay que tener en cuenta que el parámetro m controla la difusividad de los clusters. Si $m=1$, FCM se comporta como K-Means, mientras que valores mayores de m generan clusters más difusos.

En la figura 3.7 se puede ver el resultado de aplicar este método de agrupamiento.

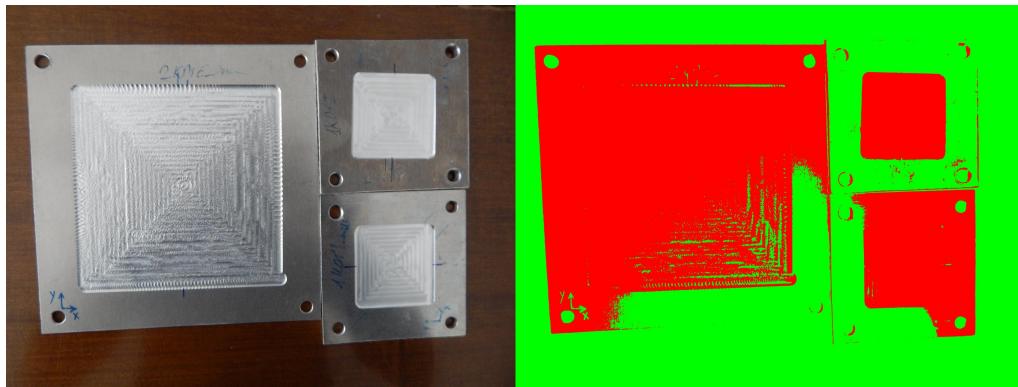


Figura 3.7: Resultado de aplicar Fuzzy C-Means

Gaussian Mixtures Models

Gaussian Mixture Models (GMM) [8] es un algoritmo de clustering basado en la idea de que los datos pueden ser modelados como una combinación de múltiples distribuciones gaussianas. Cada cluster se representa como una distribución gaussiana con su media y varianza propias. GMM es una extensión más flexible de K-Means, ya que permite que los clusters tengan diferentes formas y tamaños.

GMM es útil en aplicaciones donde los datos tienen una estructura más compleja que no puede ser modelada adecuadamente por algoritmos como K-Means. Algunas razones para utilizarlo incluyen:

1. Modelado de Datos Complejos: Es capaz de modelar clusters con formas y tamaños arbitrarios.
2. Identificación de Patrones Multimodales: Puede identificar clusters que tienen múltiples modos.
3. Análisis de Datos Probabilístico: Proporciona una probabilidad para cada punto de pertenecer a cada cluster.

El funcionamiento básico de Gaussian Mixtures Models es el siguiente:

1. Inicialización

Se establece el número de componentes gaussianas k , y se inicializan aleatoriamente los parámetros de las distribuciones: media (μ_k), covarianza (Σ_k), y pesos (π_k).

2. Cálculo de probabilidades posteriores (γ_{ik})

Se calcula la probabilidad de que cada pixel x_i pertenezca al cluster k utilizando la regla de Bayes. Esto se conoce como la responsabilidad del cluster k para el pixel x_i :

$$\gamma_{ik} = \frac{\pi_k \cdot \mathcal{N}(x_i | \mu_k, \Sigma_k)}{\sum_{j=1}^k \pi_j \cdot \mathcal{N}(x_i | \mu_j, \Sigma_j)}$$

donde:

- $\mathcal{N}(x_i | \mu_k, \Sigma_k)$ es la función de densidad de probabilidad de una distribución gaussiana con media μ_k y covarianza Σ_k .

3. Reestimación de parámetros

- Medias (μ_k):

$$\mu_k = \frac{\sum_{i=1}^n \gamma_{ik} \cdot x_i}{\sum_{i=1}^n \gamma_{ik}}$$

- Covarianzas (Σ_k):

$$\Sigma_k = \frac{\sum_{i=1}^n \gamma_{ik} \cdot (x_i - \mu_k)(x_i - \mu_k)^\top}{\sum_{i=1}^n \gamma_{ik}}$$

- Pesos (π_k):

$$\pi_k = \frac{\sum_{i=1}^n \gamma_{ik}}{n}$$

4. Repetición

Se repiten los pasos 2 y 3 hasta que los cambios en los parámetros sean menores a un umbral dado, o se alcance un número máximo de iteraciones.

En la figura 3.8 se puede ver el resultado de aplicar este método de agrupamiento.

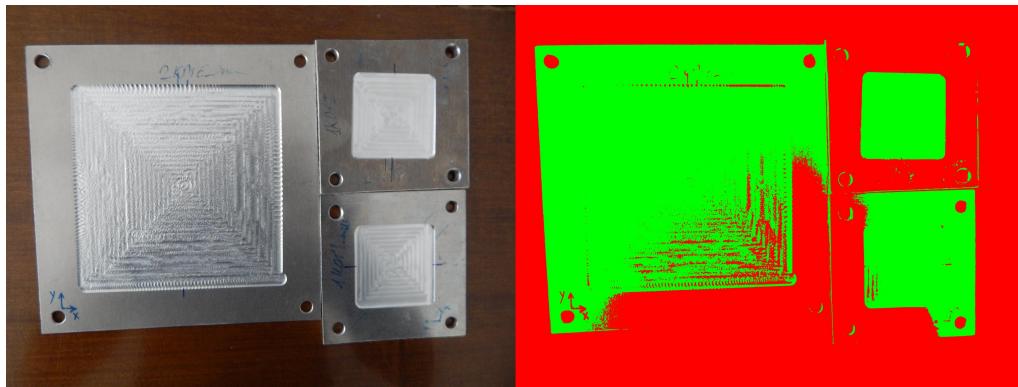


Figura 3.8: Resultado de aplicar Gaussian Mixtures Models

Información espacial

El modelo Hidden Markov Random Field (HMRF) [10] es un modelo probabilístico que combina la teoría de campos aleatorios de Markov con modelos ocultos. Se utiliza ampliamente para problemas de visión computacional, como la segmentación de imágenes. El algoritmo HMRF-EM combina el modelo HMRF con el algoritmo Expectation-Maximization (EM) para la segmentación de imágenes.

El modelo HMRF-EM (Hidden Markov Random Field - Expectation Maximization) es útil en segmentación de imágenes porque :

1. Regulariza la segmentación, reduciendo el ruido.
2. Considera relaciones entre píxeles vecinos para segmentar áreas similares.
3. El algoritmo EM ajusta parámetros para mejorar la precisión.
4. La versión Edge-Prior-Preserving mantiene bordes importantes en la imagen.

El funcionamiento básico del modelo Hidden Markov Random Field es el siguiente:

1. Inicialización

Se establece un conjunto inicial de parámetros Θ_0

$$\Theta_0 = [\mu_{initial}, \sigma_{initial}];$$

2. Cálculo de la Distribución de Verosimilitud

Se calcula la distribución de verosimilitud $P^{(t)}(y_i|x_i, \theta_{x_i})$ en la iteración t .

$$P^{(t)}(y_i|x_i, \theta_{x_i}) = G(y_i; \theta_{x_i}) = \frac{1}{\sqrt{2\pi\sigma_{x_i}^2}} \exp\left(-\frac{(y_i - \mu_{x_i})^2}{2\sigma_{x_i}^2}\right)$$

3. Estimación MAP

Se estima la configuración de etiquetas mediante el criterio de Máxima A Posteriori (MAP):

$$x^{(t)} = \arg \max_{x \in \chi} \left\{ P(y|x, \Theta^{(t)}) P(x) \right\} = \arg \min_{x \in \chi} \left\{ U(y|x, \Theta^{(t)}) + U(x) \right\}$$

donde:

- $U(y|x, \Theta)$ es la energía de verosimilitud:

$$U(y|x, \Theta) = \sum_i \left[\frac{(y_i - \mu_{x_i})^2}{2\sigma_{x_i}^2} + \ln \sigma_{x_i} \right]$$

- $U(x)$ es la energía prior:

$$U(x) = \sum_{c \in C} V_c(x)$$

- $V_c(x)$ es el potencial de clique, que en este caso se define para un par de píxeles vecinos:

$$V_c(x_i, x_j) = \frac{1}{2}(1 - I_{x_i, x_j})$$

donde:

$$I_{x_i, x_j} = \begin{cases} 0 & \text{si } x_i \neq x_j \\ 1 & \text{si } x_i = x_j \end{cases}$$

4. Actualización de Parámetros (M-step)

Se actualizan los parámetros $\Theta^{(t+1)}$ usando las distribuciones posteriores.

- Media ($\mu_l^{(t+1)}$):

$$\mu_l^{(t+1)} = \frac{\sum_i P^{(t)}(l|y_i)y_i}{\sum_i P^{(t)}(l|y_i)}$$

- Varianza (σ_l^2):

$$(\sigma_l^{(t+1)})^2 = \frac{\sum_i P^{(t)}(l|y_i)(y_i - \mu_l^{(t+1)})^2}{\sum_i P^{(t)}(l|y_i)}$$

- Posteriores ($P^{(t)}(l|y_i)$):

$$P^{(t)}(l|y_i) = \frac{G(y_i; \theta_l)P^{(t)}(l|x_{N_i}^{(t)})}{P^{(t)}(y_i)}$$

5. Repetición

Se repiten los pasos 2 a 4 hasta que la función de energía $U(y|x, \Theta) + U(x)$ converja o se alcance el número máximo de iteraciones.

En la figura 3.9 se puede ver el resultado de aplicar este método de agrupamiento.

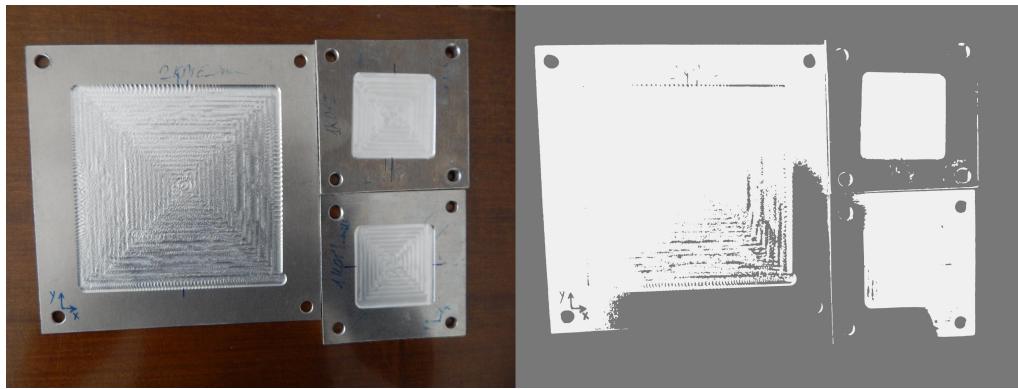


Figura 3.9: Resultado de aplicar Hidden Markov Random Field

3.4. Clasificación de resultados

La clasificación de los resultados obtenidos es un paso fundamental ya que gracias a ello podemos ver de una forma objetiva si al aplicar métodos de transformación invariante mejora los resultados obtenidos. Compararemos las imágenes, tanto la original como la invariante con una imagen “ground truth” para de esta forma observar numéricamente el acierto de cada una de ellas.

Imágenes ground truth

Las imágenes “ground truth” son segmentaciones realizadas manualmente mediante programas de edición de imágenes. Al haber sido segmentadas por un ser humano, representan una segmentación casi perfecta, que sirve como ideal al que aspiramos. Digo “casi perfecta” porque puede haber un mínimo error imperceptible a simple vista, pero que puede afectar matemáticamente, impidiendo alcanzar un 100 % de precisión.

En la figura 3.10 se puede ver el la imagen original y su correspondiente imagen “ground truth”.

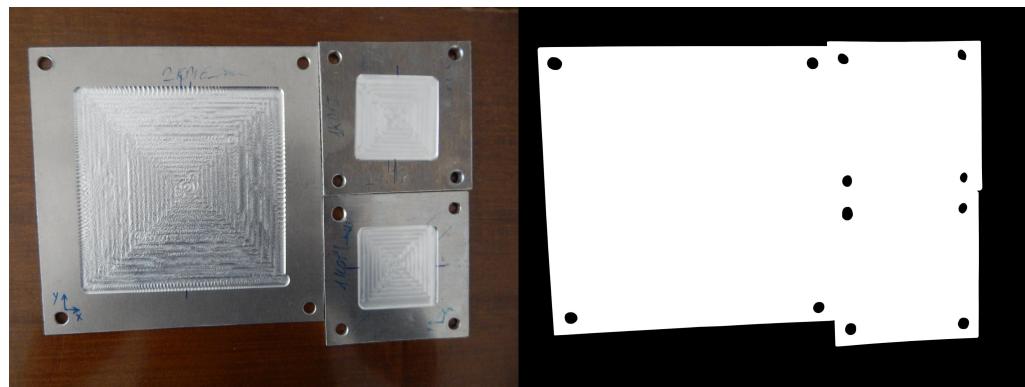


Figura 3.10: Resultado de crear la imagen “ground truth”

Porcentaje de acierto

El porcentaje de acierto nos permite evaluar la precisión de la segmentación realizada al compararla con una imagen en la que manualmente se ha segmentado la pieza del fondo (imagen ground truth). La fórmula para calcularlo entre dos imágenes es:

$$\text{coincidencias} = \sum_{i,j} \delta(\text{img1_bnw}(i,j), \text{img2_bnw}(i,j))$$

donde:

- coincidencias es el numero total de píxeles que coinciden entre las dos imágenes en blanco y negro.
- $\sum_{i,j}$ indica que se realiza una suma sobre todos los píxeles de la imagen.
- $\text{img1_bnw}(i,j)$ es el valor de cada posición (i,j) de la primera imagen
- $\text{img2_bnw}(i,j)$ es el valor de cada posición (i,j) de la segunda imagen
- δ es la función delta de Kronecker que se define como:

$$\delta(a, b) = \begin{cases} 1 & \text{si } a = b \\ 0 & \text{si } a \neq b \end{cases}$$

Finalmente la función con la que se obtiene el porcentaje de acierto entre la dos imágenes es la siguiente.

$$\text{porcentaje_coincidencia} = \left(\frac{\text{coincidencias}}{\text{numero_pixeles(img1)}} \right) \times 100$$

4. Técnicas y herramientas

Esta parte de la memoria tiene como objetivo presentar las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto. Si se han estudiado diferentes alternativas de metodologías, herramientas, bibliotecas se puede hacer un resumen de los aspectos más destacados de cada alternativa, incluyendo comparativas entre las distintas opciones y una justificación de las elecciones realizadas. No se pretende que este apartado se convierta en un capítulo de un libro dedicado a cada una de las alternativas, sino comentar los aspectos más destacados de cada opción, con un repaso somero a los fundamentos esenciales y referencias bibliográficas para que el lector pueda ampliar su conocimiento sobre el tema.

5. Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros³, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

6. Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

7. Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Bibliografía

- [1] Reflexión de la luz - Wikipedia, la enciclopedia libre — es.wikipedia.org. https://es.wikipedia.org/wiki/Reflexi%C3%B3n_de_la_luz. [Accessed 29-04-2024].
- [2] 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.
- [3] 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.
- [4] 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.
- [5] 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.
- [6] MathWorks Inc. *Fuzzy Logic Toolbox: Design and Simulate Fuzzy Logic Systems*, 2023. Version 10.3 (R2023b).
- [7] MathWorks Inc. *MATLAB: The Language of Technical Computing*, 2023. Version 10.3 (R2023b).

- [8] MathWorks Inc. *Statistics and Machine Learning Toolbox: Use fitgmdist for Fitting Gaussian Mixture Models*, 2023. Version 10.3 (R2023b).
- [9] 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.
- [10] Quan Wang. Hmrf-em-image: implementation of the hidden markov random field model and its expectation-maximization algorithm. *arXiv preprint arXiv:1207.3510*, 2012.