



UNIVERSIDAD DE DEUSTO

SEGMENTACIÓN Y CLASIFICACIÓN DE DEFECTOS SUPERFICIALES EN PIEZAS DE FUNDICIÓN DE HIERRO NODULAR

IKER PASTOR LÓPEZ

Bilbao, marzo de 2013



UNIVERSIDAD DE DEUSTO

SEGMENTACIÓN Y CLASIFICACIÓN DE DEFECTOS SUPERFICIALES EN PIEZAS DE FUNDICIÓN DE HIERRO NODULAR

Tesis doctoral presentada por
IKER PASTOR LÓPEZ
dentro del Programa de Doctorado en
INFORMÁTICA Y TELECOMUNICACIÓN

Dirigida por el
Dr. IGOR SANTOS GRUEIRO
y por el
Dr. PABLO GARCÍA BRINGAS

Autor

Co-director

Co-director

Bilbao, marzo de 2013

A mi madre.

Por levantarme siempre que me he caído.

Resumen

Los procesos productivos son uno de los indicadores más representativos del progreso de una sociedad. Es por ello, que su optimización se ha convertido en uno de los principales objetivos del ser humano. Ésta, se ha materializado, en algunos casos, como mejoras aplicadas al producto fabricado, y en otros, como avances en el propio proceso en sí.

Uno de los procesos productivos más importantes es el de la fundición. A grandes rasgos, este proceso conlleva la fusión de un material y su posterior enfriamiento en un molde en el que se le proporciona la forma deseada.

Concretamente, en esta tesis doctoral nos vamos a centrar en la fundición de hierro. Ésta, se encarga de suministrar piezas clave a otras industrias como la automovilística, la naval o la aérea entre muchas otras. Es por este motivo, que estas piezas son sometidas a estrictos controles de calidad con los que se busca eliminar los posibles defectos que puedan haber surgido a lo largo del proceso. Actualmente, esta inspección se realiza de forma manual, lo que conlleva una serie de riesgos inherentes a los operarios que realizan dicha tarea.

En esta tesis doctoral proponemos un nuevo sistema de visión artificial, que es capaz de determinar la validez de una pieza de fundición de hierro, detectando y categorizando los defectos superficiales que puedan aparecer en la misma. Para ello, hemos diseñado y construido un sistema de visión artificial, que es capaz de capturar la información superficial de las piezas, y a continuación, genera distintas representaciones de la información capturada.

Posteriormente, proponemos un método de segmentación, que partiendo de las representaciones gráficas generadas, delimita las regiones de la superficie susceptibles de contener un defecto. Además, produce una representación de las mismas en base a distintos tipos de características extraídas.

Para la detección y categorización de los defectos existentes en las regiones, planteamos un método de categorización basado en algoritmos de aprendizaje automático supervisado.

Como último aporte, planteamos una mejora del método de segmentación propuesto, y que se basa en técnicas de *clustering*. Asimismo, proponemos una mejora del método de categorización basada en algoritmos de aprendizaje automático colectivo. Con ambas mejoras, conseguimos una mejora sustancial de los métodos propuestos anteriormente.

Agradecimientos

En este pequeño apartado quiero dejar constancia de todas las personas que me han ayudado con su trabajo y apoyo, pasando a formar parte de esta tesis doctoral. Quiero aclarar que el orden que he seguido no implica mayor o menor importancia.

En primer lugar, quiero dar las gracias al Dr. Pablo García Bringas. Él, fue quien me dio la oportunidad hace ya cerca de 4 años de formar parte del laboratorio S³lab; todo un orgullo para mí. Además, quiero darle las gracias por el tiempo y el esfuerzo dedicado durante estos años de dirección, que me han ayudado a llegar a este punto de mi carrera. Muchas gracias, Pablo.

En segundo lugar, quiero dar las gracias al Dr. Igor Santos Grueiro. Él, ha sido uno de los pilares básicos de esta tesis, y un buen amigo. Además, tengo que agradecerle el haberme contagiado su entusiasmo por la ciencia, algo realmente difícil en los tiempos que corren. También quiero agradecerle el haber dedicado tanto tiempo y esfuerzo, tanto en los momentos de risas, como en los momentos de estar a punto de tirar la toalla. Muchas gracias, Igor.

En tercer lugar, quiero dar las gracias a todo el equipo humano que compone el S3lab: Félix, Mikel, Juan, Carlos, Javi, José, Borja, Iván, Aitor, Sendoa, Asier, Jaime, Xabi Cantero, Jorge, Patxi, Xabi Ugarte y Agus. Todos y cada uno de ellos demuestran cada día el valor que atesoran. Muchas gracias, a todos.

En cuarto lugar, quiero dar las gracias a mis amigos. A Raúl, porque nunca me ha dejado en la estacada y siempre ha estado ahí. A Iker, porque siempre he podido contar con él. A Javi y Jon, por convertir las clases de aikido en algo más que un arte marcial. Muchas gracias, a Raúl, Iker, Javi y Jon.

En quinto lugar, quiero dar las gracias a mi familia, sobre todo por haber estado animándome en todo momento, y en especial a mi madre, que cada día sigue enseñándome más cosas. Muchas gracias, a mi padre Aurelio, a mi madre Clara, a mi hermana Clara, a mi hermano Aurelio, a mi cuñado Lucio, a mi cuñada Eva, y a mis sobrinas Andrea y Adriana.

En sexto lugar, quiero dar las gracias a DeustoTech y a la Universidad de Deusto por haber hecho posible esta tesis doctoral. Muchas gracias.

Por último, quiero dar las gracias a Azterlan por habernos proporcionado el conjunto de piezas que hemos empleado para la experimentación, así como por la colaboración prestada a la hora de etiquetar el conjunto de datos de entrenamiento. Muchas gracias.

Índice general

Índice de figuras	xiii
-------------------	------

Índice de tablas	xvii
------------------	------

1	Introducción	3
1.1	Motivación	4
1.2	El proceso de fundición del metal	4
1.2.1	Fusión	4
1.2.2	Moldeo	5
1.2.3	Colada	5
1.2.4	Enfriamiento	6
1.2.5	Verificación	6
1.3	Defectos presentes en piezas de fundición de hierro nodular . .	7
1.3.1	Defectos ocasionados por la arena	7
1.3.1.1	Porosidad	7
1.3.1.2	Arruga	7
1.3.1.3	Vitrificación	7
1.3.1.4	Grieta	8
1.3.1.5	Inclusión	8
1.3.1.6	Hundimiento	8
1.3.1.7	Desplazamiento del molde	8
1.3.2	Defectos ocasionados por el metal	9
1.3.2.1	Rechupe	9
1.3.2.2	Precipitación de carbono	9
1.3.2.3	Malformación gráfica esferoidal	9
1.3.2.4	Formación de carburos	9

1.3.2.5	Unión fría	9
1.4	Hipótesis y objetivos	10
1.5	Metodología de la investigación	11
1.6	Arquitectura global del sistema propuesto	13
1.7	Estructura del documento	14
2	Estudio de la literatura	17
2.1	Arquitectura de un sistema de visión artificial	17
2.2	Elementos de un sistema de visión artificial	18
2.2.1	Dispositivo de captura	18
2.2.1.1	Sistemas pasivos	19
2.2.1.2	Sistemas activos	20
2.2.2	Dispositivo de automatización	23
2.2.3	Equipo de procesamiento	24
2.3	Fases de un sistema de visión artificial	25
2.3.1	Adquisición de imágenes	25
2.3.2	Procesado de imágenes	26
2.3.3	Extracción de características	26
2.3.4	Clasificación	27
2.4	Sistemas de visión artificial para la inspección de calidad . . .	27
2.4.1	Inspección de la calidad dimensional	27
2.4.2	Inspección de la calidad estructural	30
2.4.3	Inspección de la calidad operacional	30
2.4.4	Inspección de la calidad superficial	31
2.4.5	Detección de defectos superficiales mediante el análisis de textura	32
2.4.5.1	Enfoques estadísticos	33
2.4.5.2	Enfoques estructurales	35
2.4.5.3	Enfoques basados en filtro	36
2.4.5.4	Enfoques basados en modelo	37
2.5	Sumario	38
3	Arquitectura del sistema de visión artificial propuesto	41
3.1	Elementos del sistema de visión artificial propuesto	42
3.1.1	Dispositivo de captura	42
3.1.2	Dispositivo de automatización	43

3.1.3	Equipo de procesamiento	43
3.2	Conjunto de datos empleado para la validación	44
3.3	Obtención de los datos	44
3.4	Normalización de los datos	46
3.5	Representaciones utilizadas	47
3.5.1	Matriz de alturas	47
3.5.2	Mapa de alturas en escala de grises	48
3.5.3	Matriz de vectores normales	48
3.5.4	Mapa de vectores normales codificado en RGB	49
3.6	Segmentación de las imágenes	50
3.6.1	Proceso de segmentación	51
3.7	Sumario	53
4	Categorización de defectos superficiales	55
4.1	Conjuntos de características extraídas de los regiones detectadas	56
4.1.1	Características simples	56
4.1.2	Características del Best Crossing Line Profile	57
4.1.3	Características de la transformada rápida de Fourier	58
4.1.4	Características de la matriz de co-ocurrencia	60
4.2	Categorización de defectos basada en aprendizaje automático supervisado	61
4.2.1	Redes bayesianas	61
4.2.2	Máquinas de soporte vectorial	62
4.2.3	K vecinos más cercanos	64
4.2.4	Árboles de decisión	65
4.3	Validación empírica de la categorización de defectos	66
4.3.1	Metodología de evaluación	66
4.3.2	Medición de los resultados	67
4.3.2.1	Tasa de verdaderos positivos	67
4.3.2.2	Tasa de falsos positivos	68
4.3.2.3	Tasa agregada de verdaderos positivos	68
4.3.2.4	Tasa agregada de falsos positivos	68
4.3.2.5	Precisión	68
4.3.2.6	Área bajo la curva ROC	68
4.3.3	Resultados obtenidos mediante la experimentación	69
4.3.3.1	Categorización en dos fases	69

4.3.3.2	Categorización directa	86
4.4	Discusión de los resultados	93
4.5	Sumario	96
5	Optimización del método propuesto	97
5.1	Mejora propuesta para el método de segmentación	98
5.1.1	Algoritmos de <i>clustering</i>	98
5.1.1.1	El algoritmo <i>Quality Threshold</i>	98
5.1.2	Validación empírica de la mejora propuesta para el método de segmentación	100
5.1.2.1	Metodología de evaluación y medición de los resultados	100
5.1.2.2	Resultados obtenidos mediante la experimentación	101
5.2	Mejora propuesta para el método de categorización	105
5.2.1	Clasificación colectiva	106
5.2.1.1	<i>Collective IBK</i>	106
5.2.1.2	<i>Collective Forest</i>	107
5.2.1.3	<i>Collective Woods</i> y <i>Collective Tree</i>	107
5.2.1.4	<i>RandomWoods</i>	108
5.2.2	Enfoques multiclase	108
5.2.2.1	Todos contra todos	108
5.2.3	Validación empírica de la mejora propuesta para el método de categorización	108
5.2.3.1	Metodología de evaluación y medición de los resultados	109
5.2.3.2	Resultados obtenidos mediante la experimentación	110
5.3	Discusión de los resultados	123
5.4	Sumario	125
6	Conclusiones	127
6.1	Resumen de los resultados	127
6.2	Limitaciones	130
6.3	Trabajo futuro	131
6.3.1	Mejorar el hardware del sistema de visión artificial propuesto	131
6.3.2	Mejorar el método de segmentación de imágenes	132

6.3.3	Mejorar el método de categorización de defectos	132
6.4	Consideraciones finales	132
	Bibliografía	135

Índice de figuras

1.1	Arquitectura global del sistema.	13
2.1	Arquitectura de un sistema de visión artificial clásico.	18
2.2	Clasificación de los dispositivos de captura 3D.	19
2.3	Clasificación de los dispositivos de captura 3D pasivos.	20
2.4	Clasificación de los dispositivos de captura 3D activos.	21
2.5	Fases de un sistema de visión artificial.	26
2.6	Clasificación de aplicaciones de inspección de calidad.	27
2.7	Clasificación de los enfoques de análisis de textura para la de- tección de defectos superficiales propuesta.	33
2.8	Ejemplo simplificado de un histograma.	34
3.1	Prototipo del sistema de visión artificial propuesto.	42
3.2	Visualización de la matriz de alturas mediante el renderizado de una malla 3D.	48
3.3	Mapa de alturas en escala de grises.	49
3.4	Visualización de los vectores normales sobre una malla 3D. . .	49
3.5	Mapa de vectores normales codificados en RGB.	50
3.6	Mapa de normales codificado en RGB de la pieza defectuosa. .	51
3.7	Imagen obtenida mediante el método de segmentación descrito.	53
4.1	Ejemplo de clasificación mediante una red bayesiana.	62
4.2	Ejemplo de clasificación mediante una máquina de soporte vec- torial.	63
4.3	Ejemplo de clasificación mediante de k-vecinos más cercanos. .	64
4.4	Ejemplo de clasificación mediante un árbol de decisión.	65

5.1	Pseudo-código de la implementación del algoritmo QT empleado.	99
5.2	Visualización de la matriz de alturas de un modelo centroide mediante el renderizado de una malla 3D.	102
5.3	Resultados del método de segmentación en base al umbral de similitud empleado	103
5.4	Resultados obtenidos por los clasificadores colectivos empleando las características simples para la fase 1.	110
5.5	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas del <i>Best Crossing Line Profile</i> para la fase 1.	111
5.6	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas de la transformada rápida de Fourier para la fase 1.	112
5.7	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas de la matriz de co-ocurrencia para la fase 1.	113
5.8	Resultados obtenidos por los clasificadores colectivos empleando todas las características para la fase 1.	114
5.9	Resultados obtenidos por los clasificadores colectivos empleando las características simples para la fase 2.	115
5.10	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas del <i>Best Crossing Line Profile</i> para la fase 2.	115
5.11	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas de la transformada rápida de Fourier para la fase 2.	116
5.12	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas de la matriz de co-ocurrencia para la fase 2.	116
5.13	Resultados obtenidos por los clasificadores colectivos empleando todas las características para la fase 2.	117
5.14	Resultados obtenidos por los clasificadores colectivos empleando las características simples para la categorización directa. . .	118
5.15	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas del <i>Best Crossing Line Profile</i> para la categorización directa.	119
5.16	Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas de la transformada rápida de Fourier para la categorización directa.	120

5.17 Resultados obtenidos por los clasificadores colectivos empleando las características simples combinadas con las extraídas de la matriz de co-ocurrencia para la categorización directa. . . .	121
5.18 Resultados obtenidos por los clasificadores colectivos empleando todas las características para la categorización directa. . . .	122

Índice de tablas

2.1	Comparativa de los sistemas de captura de información 3D presentados.	23
3.1	Distribución de apariciones de cada tipo defecto.	44
3.2	Distribución de regiones con cada tipo de defecto.	53
4.1	Algoritmos de aprendizaje supervisados, junto con sus configuraciones.	67
4.2	Resultados obtenidos empleando las características extraídas del <i>Best Crossing Line Profile</i> para la fase 1.	70
4.3	Resultados obtenidos empleando las características extraídas de la transformada rápida de Fourier para la fase 1.	71
4.4	Resultados obtenidos empleando las características extraídas de la matriz de co-ocurrencia para la fase 1.	72
4.5	Resultados obtenidos empleando las características simples para la fase 1.	73
4.6	Resultados obtenidos empleando las características del <i>Best Crossing Line Profile</i> combinadas con las características simples para la fase 1.	74
4.7	Resultados obtenidos empleando las características de la transformada rápida de Fourier combinadas con las características simples para la fase 1.	75
4.8	Resultados obtenidos empleando las características de la matriz de co-ocurrencia combinadas con las características simples para la fase 1.	76
4.9	Resultados obtenidos empleando empleando todas las características para la fase 1.	77
4.10	Resultados obtenidos empleando las características del <i>Best Crossing Line Profile</i> para la fase 2.	78

4.11 Resultados obtenidos empleando las características de la transformada rápida de Fourier para la fase 2.	79
4.12 Resultados obtenidos empleando las características de la matriz de co-ocurrencia para la fase 2.	80
4.13 Resultados obtenidos empleando las características simples para la fase 2.	81
4.14 Resultados obtenidos empleando las características del <i>Best Crossing Line Profile</i> combinadas con las características simples para la fase 2.	82
4.15 Resultados obtenidos empleando las características de la transformada rápida de Fourier combinadas con las características simples para la fase 2.	83
4.16 Resultados obtenidos empleando las características de la matriz de co-ocurrencia combinadas con las características simples para la fase 2.	84
4.17 Resultados obtenidos empleando todas las características para la fase 2.	85
4.18 Resultados obtenidos empleando las características del <i>Best Crossing Line Profile</i> para la categorización directa.	86
4.19 Resultados obtenidos empleando las características de la transformada rápida de Fourier para la categorización directa.	87
4.20 Resultados obtenidos empleando las características de la matriz de co-ocurrencia para la categorización directa.	88
4.21 Resultados obtenidos empleando las características simples para la categorización directa.	89
4.22 Resultados obtenidos empleando las características del <i>Best Crossing Line Profile</i> combinadas con las características simples para la categorización directa.	90
4.23 Resultados obtenidos empleando las características de la transformada rápida de Fourier combinadas con las características simples para la categorización directa.	91
4.24 Resultados obtenidos empleando las características de la matriz de co-ocurrencia combinadas con las características simples para la categorización directa.	92
4.25 Resultados obtenidos empleando todas las características para la categorización directa.	93
4.26 Comparación de los mejores clasificadores para la fase 1.	94
4.27 Comparación de los mejores clasificadores para la fase 2.	94
4.28 Comparación de los mejores clasificadores para la categorización directa.	95

5.1	Resultados obtenidos en base al umbral de similitud seleccionado para el algoritmo de agrupación.	102
5.2	Distribución de regiones con cada tipo de defecto.	104
5.3	Comparación de los mejores clasificadores para la fase 1. . . .	104
5.4	Comparación de los mejores clasificadores para la fase 2. . . .	105
5.5	Comparación de los mejores clasificadores para la categorización directa.	105
5.6	Algoritmos de aprendizaje colectivo empleados, junto con sus configuraciones.	109
5.7	Comparación de los mejores clasificadores supervisados y colectivos para la fase 1.	123
5.8	Comparación de los mejores clasificadores supervisados y colectivos para la fase 2.	124
5.9	Comparación de los mejores clasificadores supervisados y colectivos para la categorización directa.	124

«No importa lo fuerte que golpeas, sino lo fuerte que pueden golpear-te, y lo que aguantas mientras avanzas. Hay que soportar sin dejar de avanzar. ¡Levántate, levántate!..»

Rocky Balboa

«El auténtico problema no es si las máquinas piensan, sino si lo hacen los hombres.»

Frederic Burrhus Skinner (1904–1990)

CAPÍTULO

1

Introducción

ACTUALMENTE, la automatización de los procesos productivos es una de las respuestas más efectivas a la demanda creciente de servicios y productos por parte de la sociedad, la cual, conlleva no sólo una optimización del producto fabricado, sino también un progreso continuo del propio proceso en sí. Esta mejora se ha traducido a su vez en una disminución de los costes productivos generados, haciendo a la empresa más competitiva en un mercado cada vez más globalizado.

Pero esta automatización implica una serie de riesgos y limitaciones tecnológicas y científicas que pueden hacerla inviable. Un ejemplo en el que podemos observar estas dificultades, es el de la visión artificial, en el que debemos sustituir el sistema óptico y el conocimiento de un ser humano para procesar la información visual obtenida.

En concreto, el objeto de estudio de esta tesis se centra en el campo de la calidad dentro de los procesos productivos, y más concretamente en la detección y clasificación de defectos superficiales en piezas de fundición de hierro nodular.

El resto de este capítulo continúa con la siguiente estructura. La Sección 1.1 describe la motivación que ha empujado a la realización de esta tesis doctoral. La Sección 1.2 describe el proceso de fundición de hierro. La Sección 1.3 describe los diferentes defectos superficiales existentes en la piezas de fundición de hierro nodular. La Sección 1.4 establece la hipótesis de esta tesis, así como los objetivos que se deben cumplir para llegar a la consecución de la misma. La Sección 1.5 especifica la metodología que se ha empleado para la

investigación. Por su parte, la Sección 1.6 muestra la arquitectura global del sistema que proponemos para cumplir con los objetivos marcados. Finalmente, la Sección 1.7 presenta la estructura general de la presente tesis doctoral.

1.1 Motivación

La fundición es uno de los procesos más antiguos de la historia de la humanidad, siendo en cada una de las épocas uno de los indicadores principales del progreso de la misma. Básicamente, consiste en fundir un metal y verterlo en el interior de un molde hasta que se solidifique con la forma deseada. Las piezas obtenidas por medio de este proceso, son básicas para otras industrias, como por ejemplo la aeronáutica, la armamentística, la naval o la automovilística. Ésto, obliga a que las piezas producidas se sometan a exhaustivos controles de calidad a fin de eliminar aquellas que puedan ser motivo de rechazo debido a la aparición de algún defecto.

Es por este motivo, que se hace necesaria la automatización de los controles de calidad del proceso de fundición, para optimizarlos y conseguir el objetivo de disminuir el riesgo que supone la utilización de una pieza defectuosa en las industrias destino.

Actualmente, el proceso de inspección de defectos se hace de forma manual. Ésto, conlleva una serie de riesgos asociados al estado anímico, fatiga y otros factores que pueden influir de forma negativa en el operador encargado de determinar la validez de la pieza inspeccionada. Estos riesgos generan la necesidad de automatizar este proceso a fin de minimizar el riesgo inherente a la condición humana del operario.

1.2 El proceso de fundición del metal

El proceso de fundición es el que se emplea para obtener a partir de un metal dado, otro muy parecido al inicial pero caracterizado por tener una forma y características distintas. Este proceso data de la antigua edad del hierro, y en esencia sigue teniendo los mismos pasos aunque ha ido incorporando distintos avances tecnológicos que lo han ido mejorando. Las etapas principales en las que se divide se especifican a continuación [Cam03][DBKK03]: (i) la fusión, (ii) el moldeo, (iii) la colada, (iv) el enfriamiento, y (v) la verificación.

1.2.1 Fusión

En este primer paso, se vierten al horno las diferentes materias primas que se van a fundir. El tamaño de los hornos dependerá del tamaño y de la cantidad de piezas que se quieren hacer. Es necesario tener en cuenta, que la temperatura

de fusión del metal es elevada, variando desde los 650°C del aluminio a los 1350°C del hierro. Para alcanzar estas temperaturas se requiere de tiempo, por ello se utilizan hornos de gran capacidad y, generalmente, que trabajen en paralelo para poder tener un suministro continuo que cubra la cantidad de metal fundido demandada para posteriores etapas.

Como ya hemos comentado anteriormente, la materia prima principal es un metal base, como por ejemplo el hierro o el aluminio, pero éste nunca se vierte en estado puro sino que se emplea una combinación lingote, chatarra y piezas defectuosas y/o sobrantes. Esta combinación básicamente se realiza con un objetivo económico, ya que reciclando chatarra y piezas que no hayan superado los procesos de calidad correspondientes, se pueden generar nuevas. Es importante destacar, que esta combinación debe realizarse empleando cantidades preestablecidas para controlar la composición química de la mezcla; de esta forma, se evita la aparición de defectos en posteriores etapas.

Para verificar la composición química de la mezcla en el horno, se realizan distintos análisis químicos con los que se determina, además de la combinación expuesta anteriormente, diferentes aditivos para conseguir que el metal resultante tenga las características deseadas. Éstas, se calculan en base a los análisis de composición del horno, y dependen en gran medida de las cantidades de cada materia prima que se esté usando.

1.2.2 Moldeo

Esta tarea es una de las más críticas de todo el proceso ya que la menor desviación que se produzca en los parámetros de fabricación, puede hacer que el molde no soporte la temperatura del metal fundido y por tanto las piezas no tengan las características geométricas deseadas. Existen materiales con los que se emplean moldes metálicos, como por ejemplo el aluminio, ya que su temperatura de fusión es inferior al metal con el que se realiza el molde, consiguiendo de esta forma que no se deshaga al verter el material fundido en su interior.

Los moldes fabricados en este paso generalmente, están compuestos por distintos tipos de arena, la cual se encuentra inicialmente almacenada en las tolvas donde se le añaden unos pocos aditivos como la *bentonita*, a fin de incrementar la fuerza a la hora de ser comprimida. Posteriormente, y después de realizar todo el proceso, la arena es devuelta a las tolvas iniciales para su posterior reutilización.

1.2.3 Colada

Una vez que se ha obtenido el metal base y el horno ha alcanzado una temperatura predeterminada, se transporta hasta la área de colada. Para ello, se utilizan recipientes móviles denominados cucharas, que son de una capacidad

inferior a la del horno. Éstas, pueden variar dependiendo de las cantidades de metal fundido que se requiera transportar en cada caso. Antes de introducirlo en el sistema de colada, se añaden los últimos aditivos para obtener las propiedades óptimas y verter el metal sobre los moldes ya preparados en la fase anterior.

En caso de que el metal lo requiera, se realizan más tratamientos sobre el caldo que se está vertiendo al molde en ese momento. Uno de los más importantes que se suele dar es la *inoculación*.

1.2.4 Enfriamiento

El siguiente paso del proceso, comienza con la mezcla ya vertida en los moldes. Aquí, se empieza a enfriar y a solidificar con la forma deseada. Una vez que el metal ha disminuido su temperatura hasta la mitad aproximadamente, ya se puede romper el molde debido a que no varía la forma geométrica de la pieza. Posteriormente, la arena se recicla en tolvas y se eliminan las partes sobrantes de las piezas, como los conductos de alimentación y mazarotas. Finalmente, al enfriarse la pieza, se granalla eliminando, de esta forma, posibles restos de arena que puedan haber quedado en la superficie.

1.2.5 Verificación

En esta última etapa, todas las piezas que han sido fabricadas pasan unos exhaustivos controles de verificación. Esto se debe en gran parte a que estas piezas serán elementos clave en otras industrias, como la automovilística o la aérea entre muchas otras.

Esta tarea de verificación no se realiza sobre todas las piezas producidas, ya que sería muy costoso en cuanto a tiempo y en cuanto a coste económico, por lo que se cogen muestras representativas de cada colada y se realizan dos tipos de verificación que señalamos a continuación.

- **Exterior.** Se comprueban los defectos que puedan existir en la superficie de la pieza mediante una inspección ocular. Esta inspección se realiza sobre lotes completos.
- **Interior.** Para realizar esta inspección existen diversos métodos como los rayos x, los ultrasonidos o la tomografía; de esta forma, se buscan defectos que puedan haber surgido en el interior de la pieza y que provocan una pérdida en las propiedades mecánicas de la misma.

Una vez de que se han realizado todos estos procesos, las piezas obtenidas están listas para su comercialización.

1.3 Defectos presentes en piezas de fundición de hierro nodular

Durante la última etapa del proceso de fundición (véase sección 1.2.5), se aplican estrictos controles de calidad para detectar posibles defectos que hayan podido surgir a lo largo de todo el proceso y que puedan suponer una disminución de las propiedades de la pieza. Como ya hemos explicado, estas piezas son clave para otras industrias, por lo que hasta el más mínimo defecto las puede hacer inválidas. En esta sección vamos a ver los defectos más comunes divididos en grupos, dependiendo de su origen [Rao00][Cam03]. Éstos, son: (i) los defectos ocasionados por la arena y (ii) los defectos ocasionados por el metal.

1.3.1 Defectos ocasionados por la arena

El origen de este grupo de defectos está en el molde de arena en el que se ha vertido la mezcla fundida. Existen varios tipos, pero todos ellos se identifican generalmente por ser fácilmente visibles, por lo que no requieren de ningún sistema o equipo adicional para su detección. A continuación especificamos los tipos más representativos [DKM65][Par71], entre los que se encuentran: (i) las porosidades, (ii) las arrugas, (iii) las vitrificaciones, (iv) las grietas, (v) las inclusiones, (vi) los hundimientos, y (vii) los desplazamientos de molde.

1.3.1.1 Porosidad

La porosidad se representa por pequeñas burbujas de aire que generalmente se encuentran en la superficie de la pieza. Existen distintos tipos de porosidad dependiendo de la superficie donde se encuentre, de su contenido en nitrógeno o del tamaño que tenga.

1.3.1.2 Arruga

Este defecto identifica porque aparecen surcos de profundidad reducida en la superficie de la pieza y, generalmente, suele detectarse en superficies horizontales o convexas, además de, en superficies con un grosor estrecho. Además, las aleaciones con una fuerte tendencia a la oxidación son más propensas a sufrir este tipo de defecto.

1.3.1.3 Vitrificación

Se trata de una película de arena que se ha adherido a la superficie de la pieza. Su eliminación en algunos casos no es posible debido a que se funde

junto con el metal vertido al molde. Generalmente aparece en las zonas en las que el molde soporta más calor.

1.3.1.4 Grieta

Este defecto se caracteriza por una hendidura alargada. Dependiendo de si la superficie tiene restos de óxido se tratará de una rotura surgida con el metal caliente o de una surgida con el metal en frío.

- **Rotura en caliente.** Son fisuras con un trazado irregular y de una profundidad variable. En la parte interna existe oxidación y se presenta en numerosas ocasiones en las últimas zonas en solidificar.
- **Rotura en frío.** Son hendiduras de bordes afilados, y normalmente atraviesan toda la sección de la pieza. A diferencia de la rotura en caliente, aparece en zonas que soportan tensiones.

1.3.1.5 Inclusión

El origen de la inclusión proviene de la adhesión de algún material a la pieza en alguna de las etapas del proceso de fundición. A continuación, indicamos los distintos tipos de inclusión que podemos encontrar [Hei01].

- **Arena.** Su forma es irregular y pueden aparecer en cualquier lugar de la superficie de la pieza. Generalmente, están asociadas a las inclusiones de escoria.
- **Escoria.** Su forma es irregular y preferentemente suele encontrarse en las zonas superiores de la pieza. La razón principal por la que se forma es una reacción entre el material fundido y la arena.

1.3.1.6 Hundimiento

Se trata de una depresión que en numerosas ocasiones es paralela a la superficie y cuyo acabado es similar al del resto de la superficie. Generalmente, sucede cuando se desprende algún fragmento de arena de una parte del molde [Kot72].

1.3.1.7 Desplazamiento del molde

Este tipo de defecto se suele dar cuando se está utilizando un molde compuesto por más de una pieza y una de estas partes del molde se desplaza de su posición. Su presencia se podrá ver de forma clara en la junta de unión de los dos moldes o en las cavidades de machos¹ [Kot72].

¹Pieza formada por un compuesto de arena y resina, que se inserta dentro del molde para generar huecos en la pieza.

1.3.2 Defectos ocasionados por el metal

El origen de este grupo de defectos está relacionado principalmente con las propiedades del metal. Como ocurre con los defectos ocasionados por la arena, existen varios tipos de defecto. A continuación especificamos los tipos más representativos, entre los que se encuentran: (i) los rechupes, (ii) las precipitaciones de carbono, (iii) las malformaciones gráficas esferoidales, (iv) las formaciones de carburos, y (v) las uniones frías.

1.3.2.1 Rechupe

Generalmente, se presenta bajo la forma de cavidades abiertas o cerradas y en zonas que suelen ser las últimas en solidificar. El motivo de su presencia en estas zonas es principalmente la dificultad que tienen para evacuar el calor [DKM65].

1.3.2.2 Precipitación de carbono

Una de las características más significativas de este tipo de defecto es que no sigue una morfología en concreto, aunque en gran parte de las ocasiones suelen aparecer en forma de rosetones grisáceos. Generalmente, viene dado por una excesiva presencia de inoculante en la mezcla.

1.3.2.3 Malformación gráfica esferoidal

Las malformaciones gráficas esferoidales vienen dadas, principalmente, por la morfología del grafito presente en el metal. Se pueden clasificar en distintos tipos dependiendo de su morfología [LJH63][Jay80]. Entre estos tipos destacan el *chunky*, el *estallado*, el *spiky*, el *vermicular* y la flotación de grafito.

1.3.2.4 Formación de carburos

Se trata de uno de los defectos más problemáticos, ya que influye muy negativamente en las propiedades mecánicas de la pieza fundida. Se presenta como formaciones brillantes e irregulares con bordes redondeados y generalmente ocurre en piezas con secciones pequeñas, lo que implica una alta velocidad de solidificación.

1.3.2.5 Unión fría

Este defecto no suele darse en numerosas ocasiones, lo cual lo hace peculiar. Principalmente, ocurre cuando la temperatura de la colada es relativamente baja, y hace que la solidificación de varias partes de la pieza se realice en distinto tiempo. Su forma es muy parecida a la de una grieta, pero la característica

que la diferencia de esta es que las aristas en los alrededores del defecto aparecen con los bordes redondeados. Por otro lado, la superficie suele presentar un acabo superficial correcto.

1.4 Hipótesis y objetivos

«Es posible, empleando técnicas de segmentación en imágenes y herramientas de aprendizaje automático, determinar la validez de una pieza de fundición de hierro nodular, detectando e identificando diferentes tipos de defectos superficiales.»

Para probar la hipótesis planteada, a continuación especificamos una serie de objetivos generales. Éstos, a su vez, los hemos desglosado en objetivos específicos para validarlos de una forma óptima.

Objetivo principal 1. *Desarrollar y evaluar un detector y categorizador de defectos superficiales para piezas de fundición de hierro nodular.*

A continuación planteamos varios objetivos específicos con los que buscamos cumplir el objetivo principal descrito.

Objetivo específico 1. *Construir un sistema de visión artificial para la captura de los datos relativos a la superficie de las piezas de fundición.*

Objetivo específico 2. *Desarrollar y evaluar un detector de zonas potencialmente defectuosas basado en técnicas de segmentación de imágenes.*

Objetivo específico 3. *Desarrollar y evaluar un categorizador de defectos superficiales basado en técnicas de aprendizaje automático.*

El primer objetivo específico es construir un sistema de visión artificial, que sea capaz de capturar la información superficial de las piezas de fundición. Además, este sistema se debe caracterizar por cumplir los requisitos establecidos para poder trabajar en un entorno con condiciones adversas como es una fundición de hierro grafitico esferoidal.

El segundo objetivo específico es desarrollar y evaluar un método de segmentación que sea capaz de delimitar las regiones potenciales de la superficie que pueden contener un defecto superficial. Este método, se basará en la combinación de distintas técnicas de procesamiento de imagen.

Por último, el tercer objetivo específico es desarrollar y evaluar un categorizador de regiones, que extrayendo uno o varios conjuntos de características

de las mismas sea capaz mediante técnicas de aprendizaje automático de determinar el tipo de defecto existente.

A fin de lograr estos objetivos específicos, hemos definido una serie de objetivos operacionales, que indican las tareas a realizar. Los objetivos operacionales que hemos marcado son los siguientes:

- Objetivo Operacional 1.** *Diseñar, implementar y evaluar un sistema de visión artificial completo, que sea capaz de capturar la información superficial de las piezas a examinar.*
- Objetivo Operacional 2.** *Desarrollar un método para generar representaciones normalizadas de la información.*
- Objetivo Operacional 3.** *Desarrollar y evaluar un método de segmentación de imágenes, que indique las regiones de la superficie que puedan ser susceptibles de contener un defecto.*
- Objetivo Operacional 4.** *Extraer y evaluar las características más representativas de las regiones segmentadas para determinar su importancia en la categorización.*
- Objetivo Operacional 5.** *Desarrollar y evaluar un categorizador de regiones para determinar la validez de la pieza.*

1.5 Metodología de la investigación

Para conseguir los objetivos definidos, emplearemos la siguiente metodología de investigación compuesta por las fases que se enumeran a continuación: (i) la identificación de la problemática o de los retos a superar, (ii) la adquisición del conocimiento, (iii) la división de los retos a superar, (iv) el análisis e interpretación de los resultados globales, y (v) la difusión de los resultados.

- 1. Identificación de la problemática o de los retos a superar.** Durante este primer paso de la metodología de investigación escogida, identificaremos de una forma concreta y exacta el problema al que queremos dar solución, estableciendo de una forma clara los límites de la misma.
- 2. Adquisición de conocimiento.**
 - (a) En primer lugar será necesario adquirir una base de conocimiento a nivel global en el tema que se va a estudiar. Con este conocimiento conseguiremos tener un enfoque más genérico del tema que permitirá una mayor profundización en fases posteriores.
 - (b) En segundo lugar, y una vez obtenido una base de conocimiento global, procederemos a una adquisición de conocimiento relativo a un área más específica.

3. **División de los retos a superar.** Siguiendo con el concepto conocido como *Divide y Vencerás*, buscamos dividir la problemática que hemos identificado en el primero de los pasos de la metodología. De esta forma, conseguiremos generar diversos problemas de menor tamaño y complejidad a partir de uno de mayor, lo que a su vez conllevará una solución de menor complejidad. En este caso en concreto, y debido a que la solución final contempla más de un área de estudio, esta fórmula de trabajo está completamente justificada. Por otro lado, queremos destacar, que esta fase es uno de los puntos más críticos de la metodología que planteamos, ya que si no realizamos la división de la problemática de forma correcta, pondríamos en peligro la búsqueda de la solución global inicialmente definida. A continuación, especificamos las distintas fases en las que nos hemos basado para realizar esta división de la problemática.
- (a) Adquisición de un conocimiento más específico del área en concreto. Desde el momento en el que tengamos bien definidos y delimitados los diferentes problemas a los que queremos dar solución, procederemos a realizar un estudio, profundizando más en el área a tratar.
 - (b) Definición del experimento. En este punto, tenemos que realizar una definición formal sobre lo que vamos a investigar, además de las técnicas que emplearemos para ello. Como resultado de este paso, obtendremos la instrumentación, que será necesaria para la realización del experimento, así como para la medición de los resultados del mismo.
 - (c) Evaluación de los resultados parciales. En este apartado vamos a recoger los datos que obtengamos como resultado de la experimentación.
 - (d) Análisis de los resultados parciales. Por último y después de tener los resultados generados a partir de la experimentación definida, tenemos que analizarlos en base a los instrumentos que hemos definido con anterioridad, realizando de esta forma una evaluación de los mismos que solucionen el problema planteado.
4. **Análisis e interpretación de los resultados globales.** Una vez llegados este punto de la metodología, partimos con la premisa de que todas las divisiones de la problemática que hemos planteado, tienen ya solución. De esta forma, fusionando las distintas respuestas dadas a las pequeñas problemáticas encontradas, conseguiremos atajar la problemática global planteada en un primer momento.
5. **Difusión de los resultados.** Como último paso de la metodología, daremos a conocer los resultados que hemos ido obteniendo a lo largo de la experimentación realizada a la comunidad científica. De esta forma,

iremos compartiendo todo el conocimiento adquirido durante la investigación. Pero además de esto, iremos validando y contrastando en todo momento la investigación que vayamos realizando.

1.6 Arquitectura global del sistema propuesto

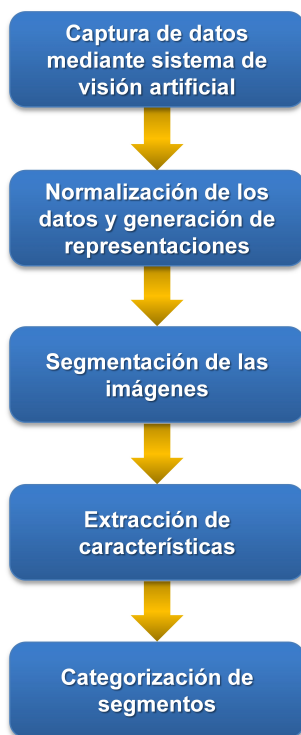


Figura 1.1: Arquitectura global del sistema.

En la Figura 1.1 podemos observar la arquitectura global que planteamos para el sistema propuesto. En concreto, se compone de las siguientes fases que iremos especificando en esta sección: (i) captura de los datos mediante un sistema de visión artificial; (ii) normalización de los datos y generación de las representaciones necesarias; (iii) segmentación de las imágenes; (iv) extracción de características para la representación de las regiones potencialmente defectuosas; y finalmente, (v) categorización de los segmentos en base a las características previamente obtenidas.

1. **Captura de los datos mediante un sistema de visión artificial.** En primer lugar tenemos que obtener los datos que van a ser la entrada de

nuestro sistema. Para ello, debemos diseñar un sistema de visión artificial, que cumpla con los requisitos de la problemática planteada, y nos ayude a obtener la información necesaria para los posteriores procesos que se van a llevar a cabo.

2. **Normalización de los datos y generación de las representaciones necesarias.** Una vez capturados los datos, debemos realizar el procesamiento necesario de los mismos para convertirlos en datos útiles para el sistema. Por ello, en primer lugar debemos normalizar los datos obtenidos, eliminando la información no relevante. Posteriormente, generaremos las representaciones que emplearemos en posteriores fases de la arquitectura.
3. **Segmentación de las imágenes.** Después de obtener las representaciones, eliminaremos las zonas que no presenten información relevante para el resto del proceso. De esta forma, simplificamos la información de entrada a los procesos posteriores del sistema.
4. **Extracción de características.** Para representar la información de las regiones de interés detectadas, es necesario realizar un estudio de las características más relevantes. Posteriormente, las extraeremos y construiremos con ellas una representación que identifique estas regiones de la manera más inequívoca posible.
5. **Categorización de los segmentos.** La representación que hayamos obtenido con las características extraídas, será la entrada para el proceso de categorización de las regiones. Es por este motivo, que esta información deberá contener la menor cantidad de ruido posible.

1.7 Estructura del documento

La tesis doctoral que presentamos a lo largo de este documento, se encuentra organizada en los siguientes capítulos: (i) introducción, (ii) estudio de la literatura, (iii) arquitectura del sistema de visión artificial propuesto, (iv) detección y categorización de defectos superficiales, (v) optimización de los métodos de detección y categorización, y (vi) conclusiones.

1. **Introducción.** En este capítulo establecemos las bases del contexto necesario para el desarrollo del resto de capítulos de la tesis. Para ello, comenzamos con la motivación que nos ha empujado a realizar el presente estudio. Continuamos con la explicación de las distintas tareas que componen el proceso de una fundición de metal. En la siguiente sección realizamos un estudio completo de los diferentes defectos superficiales, que se pueden identificar en las piezas de hierro nodular. A continuación proseguimos con los retos que buscamos superar con esta tesis doctoral.

Junto a todo esto, establecemos la hipótesis fundamental en la que nos basamos para realizar este trabajo, así como los objetivos que buscamos cumplir. Además, establecemos la arquitectura global de la solución propuesta. Finalmente, exponemos la metodología que hemos seguido para realizar la investigación de esta tesis doctoral.

2. **Estudio de la literatura.** En este capítulo realizamos un exhaustivo estudio de la literatura acerca de los sistemas de visión artificial existentes. Para ello, comenzamos planteando la arquitectura básica de un sistema de visión artificial. Posteriormente, planteamos las distintas alternativas que existen actualmente para cubrir las necesidades de cada uno de los componentes de un sistema de visión artificial. Además de esto, enumeramos las fases que componen el proceso de un sistema de visión, estableciendo los diferentes pasos que se realizan. Finalmente, concluimos con un completo estudio de los diferentes tipos de sistemas de visión artificial que han sido aplicados a la inspección de la calidad en diferentes procesos industriales, viendo una gran variedad de ejemplos al respecto.
3. **Arquitectura del sistema de visión artificial propuesto.** En este capítulo vamos a diseñar y construir el sistema de visión artificial, determinando los elementos que lo van a formar, y qué tareas van a realizar cada uno de ellos. Además, describimos las tareas que llevamos a cabo para capturar los datos de la superficie de las piezas. Posteriormente, explicamos el proceso que transforma estos datos en información útil para el sistema, eliminando el ruido que haya podido aparecer, y generando las representaciones necesarias para los procesos posteriores. A continuación, describimos el método de segmentación que proponemos para detectar las zonas de interés que pueden contener algún tipo de defecto. Por último, exponemos el conjunto piezas con las que vamos a trabajar para realizar los distintos experimentos llevados a cabo en esta tesis.
4. **Detección y categorización de defectos superficiales.** En este capítulo comenzamos especificando los distintos conjuntos de características que vamos a extraer de las regiones marcadas por el método de segmentación expuesto anteriormente. Además, explicaremos los algoritmos de aprendizaje automático supervisado que vamos a utilizar para la categorización de dichas regiones. Posteriormente, plantearemos la metodología de experimentación y de evaluación que seguiremos. Finalmente, evaluaremos los resultados arrojados por los experimentos planteados, obteniendo de esta forma los mejores resultados posibles.
5. **Optimización de los métodos de detección y categorización.** En este capítulo presentaremos mejoras para los métodos propuestos anteriormente. En primer lugar, presentaremos y evaluaremos la mejora propuesta para el método de segmentación y, posteriormente, haremos lo propio con la mejora propuesta para el método de categorización.

6. **Conclusiones.** Finalmente, en este capítulo presentaremos las conclusiones extraídas de la investigación llevada a cabo a lo largo de esta tesis doctoral. En concreto, evaluaremos los resultados obtenidos y validaremos la hipótesis fundamental planteada. Además, expondremos las limitaciones del resultado de la investigación realizada y plantearemos la líneas de trabajo futuro.

«El experimentador que no sabe lo que está buscando no comprenderá lo que encuentra.»

Claude Bernard (1813–1878)

CAPÍTULO

2

Estudio de la literatura

TRADICIONALMENTE, los procesos de control de calidad que implican una inspección visual se han llevado a cabo por operarios humanos con un determinado nivel de formación y experiencia [MGS98]. Esto se debe principalmente a que un ser humano puede desempeñar mejor que un sistema automatizado ciertas tareas aunque, por el contrario, suele ser más lento realizándolas y es sensible a la fatiga provocada por las mismas. Otro de los inconvenientes que implica es la dificultad que conlleva la búsqueda y mantenimiento de operarios expertos, ya que generalmente necesitan un tiempo sustancial para adquirir el conocimiento necesario que les permita desarrollar de forma correcta estas tareas. En el caso de la inspección visual dentro de los procesos de control, la dificultad reside principalmente en el carácter repetitivo de la tarea, que la convierte en tediosa para el operario o incluso peligrosa, dependiendo del entorno industrial en el que se encuentre. Ejemplos de este tipo de cuestión se pueden encontrar en las labores de inspección en la industria nuclear o en la industria química. Es en estas situaciones en las que los sistemas de visión artificial están reemplazando de forma progresiva a la inspección manual [MPZ⁺03].

2.1 Arquitectura de un sistema de visión artificial

Los sistemas de visión artificial generalmente siguen una arquitectura predefinida (véase la Figura 2.1), pudiendo variar los elementos que los componen dependiendo de la tecnología que se necesite emplear en cada caso.

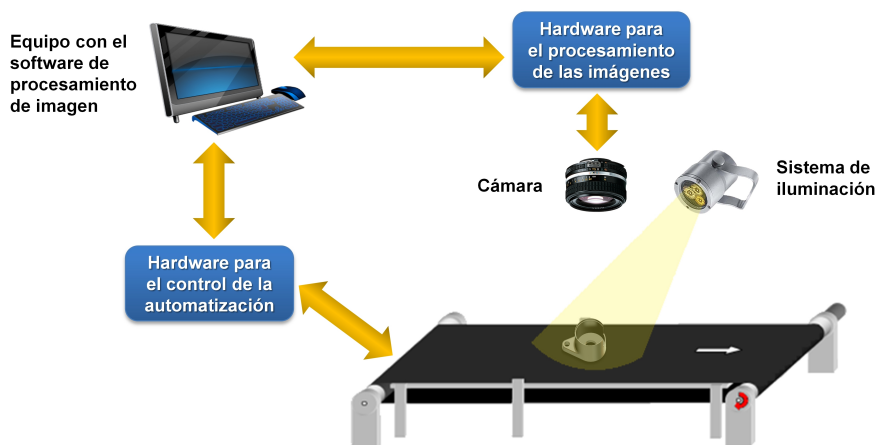


Figura 2.1: Arquitectura de un sistema de visión artificial clásico (adaptación de la versión de Malamas et ál. [MPZ⁺03]).

2.2 Elementos de un sistema de visión artificial

Como se muestra en la Figura 2.1, un sistema de visión artificial está compuesto por diversos elementos que se describen a continuación: la sección 2.2.1 trata sobre los dispositivos de captura, que son los encargados de capturar la información con la que posteriormente trabajará el sistema; la sección 2.2.2 expone los dispositivos de automatización, para la automatización los movimientos requeridos por el sistema; y finalmente la sección 2.2.3 trata sobre el equipo de procesamiento, que está compuesto por el hardware y el software necesario para la gestión operativa de todo el conjunto.

2.2.1 Dispositivo de captura

Este elemento es el encargado de obtener la información de entrada al sistema. La elección del dispositivo a emplear va a depender en cada caso de las características de los objetos a examinar, así como de las de los defectos a inspeccionar [PO03]. En los últimos años, los dispositivos más utilizados en la literatura se han centrado en la captura de información tridimensional de la superficie a examinar, dejando progresivamente de lado los dispositivos de captura 2D. Por ello, en este estudio nos vamos a centrar en los diferentes dispositivos 3D existentes.

Isgro et ál. [IOV05] proponen una clasificación para los dispositivos de captura de información 3D, en la que se diferencian dos grupos (véase la Figura 2.2): los sistemas pasivos, que son aquéllos que no emplean ninguna energía adicional a la luz ambiente o a un sistema de iluminación propio simple, para

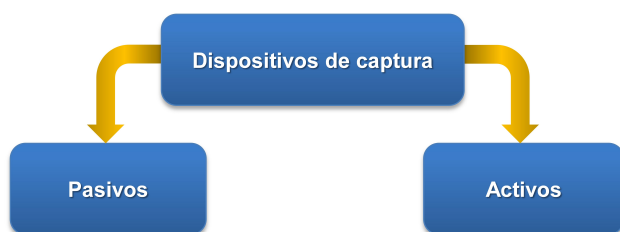


Figura 2.2: Clasificación de los dispositivos de captura 3D propuesta por Isgro et ál. [IOV05].

potenciar la información capturada por el sensor; y los sistemas activos, que capturan la información con un sensor, empleando la proyección de algún tipo de energía, ya sea láser o un patrón de iluminación predefinido.

2.2.1.1 Sistemas pasivos

Los sistemas pasivos son aquéllos que no emplean ningún tipo de energía que ayude a la captura de información por parte del sensor. Esto no implica que la escena no pueda estar iluminada por una luz ambiente, o incluso por un sistema simple de iluminación. Pernkopf y O'Leary [PO03] proponen una clasificación de los sistemas de iluminación en la que se distinguen tres tipos distintos: (i) iluminación frontal, que proyecta la luz sobre la parte frontal de la escena; (ii) iluminación trasera, en la que la fuente de luz se proyecta sobre la parte posterior de la escena, y se usa principalmente para la inspección de la silueta de objetos opacos, así como para la inspección de objetos transparentes; y (iii) la luz estructurada, que dependerá en gran medida de la escena y que se emplea generalmente para obtener información tridimensional de la misma.

Bi y Wang [BW10] completan esta clasificación agrupando los distintos dispositivos en base a la técnica que éstos emplean para la captura de la información. Por un lado se plantean los siguientes tres grupos de sistemas pasivos (véase la Figura 2.3): (i) los dispositivos que obtienen la forma mediante el movimiento, que emplean secuencias de movimiento del objeto a inspeccionar, con desplazamiento del propio objeto o de la cámara; (ii) los dispositivos basados en visión estereoscópica, que se caracterizan por emplear más de una cámara trabajando de forma concurrente; y, por último, (iii) los dispositivos que obtienen la forma mediante el sombreado, que se basan en la captura de un objeto iluminado desde distintos ángulos.

Dentro de los sistemas pasivos podemos encontrar los siguientes tipos, dependiendo de la técnica que emplean para la obtención de la información.

- **Obtención de la forma mediante el movimiento.** El funcionamiento



Figura 2.3: Clasificación de los dispositivos de captura 3D pasivos propuesta por Bi y Wang [BW10].

de este tipo de sistemas se basa principalmente, como su propio nombre indica, en la generación de una imagen 3D, para lo que se emplea el movimiento. En otras palabras, se emplean secuencias de movimiento del objeto que se está inspeccionando [STD09]. Para conseguir estas secuencias del movimiento, se pueden seguir dos enfoques distintos. El primero de ellos es el del movimiento del propio objeto. En el segundo enfoque no es el objeto lo que se mueve, sino que es la cámara; de esta forma se consigue análogo resultado. La determinación de qué enfoque se utiliza para generar las secuencias de movimiento dependerá de las características del objeto y de la cámara empleada.

- **Visión estereoscópica.** La visión estereoscópica se basa en el mismo principio de funcionamiento que el sistema visual humano. Para ello, se emplean dos o más cámaras, cada una de ellas enfocando al objeto desde un lugar distinto. Con todas las imágenes que se obtienen de forma concurrente, se genera un modelo 3D del objeto mediante triangulación [STD09].
- **Obtención de la forma mediante el sombreado.** El último de los tres tipos de sistemas pasivos se basa en la captura de varias imágenes de un mismo objeto, cambiando en cada una de ellas las condiciones de iluminación empleadas. Para conseguir los cambios en la iluminación de la escena, generalmente, se suelen emplear sistemas de iluminación estructurada [STD09].

2.2.1.2 Sistemas activos

Los sistemas activos son aquéllos que proyectan algún tipo de energía sobre el objeto o la escena a escanear, detectando posteriormente mediante sensores, generalmente ópticos, las distancias existentes [BW10]. Los diferentes tipos de energía a emplear en cada caso dependerán en gran medida de la técnica a utilizar, pero normalmente suelen ser patrones de luz, pulsos de sónar o láser.

De la misma forma que ocurre con los sistemas pasivos, Bi y Wang [BW10] plantean la siguiente división de los dispositivos activos en tres grupos (véase la Figura 2.4): (i) los basados en triangulación, cuyo funcionamiento se basa en la emisión de una energía, generalmente láser que posteriormente es capturada por un sensor óptico, calculándose de este modo la distancia a cada uno de los puntos de la superficie proyectada; (ii) los denominados tiempo de vuelo, que son aquéllos cuyo funcionamiento se basa en el mismo principio que un radar, esto es, en la emisión de pulsos láser y la posterior recogida de su incidencia en la superficie a inspeccionar; y, por último, (iii) los basados en interferimetría, que emplean una serie de patrones que se proyectan de forma periódica sobre la superficie, lo que permite recoger posteriormente la información de cada uno de ellos y uniéndola para generar la información de la superficie completa.



Figura 2.4: Clasificación de los dispositivos de captura 3D activos propuesta por Bi y Wang [BW10].

Dentro de los sistemas activos podemos encontrar los siguientes tipos, dependiendo de la técnica que emplean para la obtención de la información.

- **Basados en triangulación.** Los sistemas pertenecientes a esta categoría se basan en el principio de triangulación activa [SRR⁺03]. Estos sistemas están compuestos por un emisor que, generalmente, proyecta un haz de energía sobre el objeto y por un sensor que calcula la profundidad en cada punto, basándose en la incidencia del haz anteriormente proyectado. Existen distintos tipos, dependiendo del tipo de energía que se proyecte, siendo uno de los más empleados el láser, y de la forma empleada para la proyección, pudiendo proyectarse un solo punto a la vez, o una línea de puntos, también conocida como perfil. La principal ventaja de este tipo de sistemas es la precisión que se consigue con ellos, siendo uno de los sistemas que más se utilizan en la industria para el control de calidad de superficies, así como para la realización de distintas mediciones en las mismas [STD09].
- **Tiempo de vuelo.** Los sistemas tiempo de vuelo o de pulso láser, se basan en la medición de distancias empleando en el principio de funcionamiento de un radar. Para ello, se emplea un emisor que envía pulsos láser

sobre la superficie del objeto, incidiendo y retornando hasta un receptor. Posteriormente, se mide el tiempo entre la transmisión y la recepción, obteniendo la información de profundidad de cada uno de los puntos del objeto [STD09]. Uno de los principales inconvenientes de este tipo de sistemas es la precisión que manejan, generalmente de centímetros y que lo hacen inviable para ciertos tipos de aplicaciones [BW10].

- **Basados en interferiometría.** Los métodos de interferiometría proyectan una serie de patrones de luz sobre la superficie del objeto. Estos patrones, diferentes entre sí, se proyectan siguiendo una secuencia determinada se capturan una o varias imágenes por cada uno de ellos. Posteriormente, las imágenes obtenidas de cada uno de los patrones proyectados se emplean para reconstruir el modelo tridimensional de la superficie escaneada [STD09]. Generalmente, estos patrones se emiten mediante un proyector y se recogen mediante una cámara. La resolución con la que trabajan este tipo de sistemas depende en gran medida de la resolución del proyector empleado, así como de la de la cámara. En los últimos años y debido a los grandes avances en el dispositivos hardware, la resolución ha aumentado en gran medida [BW10].

En la Tabla 2.1, se presenta una comparativa con las fortalezas y debilidades de cada uno de los sistemas expuestos a lo largo de esta sección. Esta tabla es una adaptación de la presentada por Sansoni et ál. [STD09]. En resumen, podemos observar que los sistemas pasivos son más baratos que los activos, pero a su vez presentan una precisión menor. Otra de las cuestiones importantes a destacar es que los sistemas pasivos son más sensibles a las condiciones de los entornos en los que trabajan, lo que implica tener un mayor control de las mismas. Profundizando más, podemos observar que los sistemas de obtención de forma mediante el movimiento son baratos, sin embargo, por el contrario ofrecen una precisión pobre y su rendimiento está muy condicionado por el entorno en el que trabajan. Esto mismo sucede con la visión estereoscópica, salvo que en este caso podemos llegar a conseguir una mayor precisión. En los dispositivos basados en la obtención de forma mediante sombreado ocurre algo similar, pudiendo adaptar la cantidad de datos recogidos a la capacidad de cómputo del equipo de procesamiento utilizado. Los dispositivos basados en triangulación por su parte presentan una tasa de datos muy superior a la de cualquier sistema pasivo, y además, no son sensibles a la luz ambiente del entorno en el que trabajan, aun así, requieren de estrictas medidas de seguridad lo amplía más si cabe su coste. Otro de los inconvenientes que presentan es el de las zonas de oclusión¹ que ralentiza su funcionamiento ya que se tienen que realizar más escaneos en distintas direcciones de la misma superficie. En cuanto a los dispositivos de tiempo de vuelo, destacan por una alta tasa

¹Una zona de oclusión es una parte de la superficie que el dispositivo de captura no es capaz de recoger. En el caso de los dispositivos basados en triangulación láser son aquellas zonas en las que el sensor óptico no es capaz de ver el haz de láser proyectado sobre la superficie.

de adquisición de datos, pero reduciendo a su vez la precisión con respecto a estos. Por último, los sistema basados en interferiometría presentan una alta precisión, pero su aplicación se limita en muchos casos a superficies con una baja complejidad geométrica.

2.2.2 Dispositivo de automatización

En esta sección presentamos un estudio de los distintos dispositivos presentes en la literatura durante los últimos años para la automatización de sistemas de visión artificial aplicada a la inspección. La elección del tipo de dispositivo o dispositivos empleados en cada caso dependerá de las características del

Tabla 2.1: Tabla comparativa de los distintos sistemas de captura de información 3D presentados (adaptación de la versión de Sansoni et ál. [STD09].)

Tipo	Fortalezas	Debilidades
Obtención de forma mediante movimiento	-Bajo coste	-Rendimiento dependiente de la luz ambiente -Baja precisión
Visión estereoscópica	-Simple y bajo coste -Alta precisión en problemas concretos	-Demanda computación -Poca densidad de datos -Limitado a escenas bien definidas -Tasa baja de adquisición de datos
Obtención de forma mediante sombreado	-Bajo coste -Demanda limitada por capacidad de cómputo	-Baja precisión
Basados en triangulación	-Relativamente simple -Rendimiento independiente de la luz ambiente -Tasa alta de adquisición de datos	-Condiciones de seguridad -Rango y volumen de medición limitados -Pérdida de información por zonas de oclusión -Alto coste
Tiempo de vuelo	-Rango y volumen de medición altos -Buena tasa de adquisición de datos -Rendimiento independiente de la luz ambiente	-Precisión inferior a la triangulación -Alto coste
Basados en interferiometría	-Precisión de micras	-Alto coste -Aplicación limitada en sistemas industriales -Limitado a superficies casi planas

sistema de captura con el que se va a trabajar, así como de las características del entorno en el que se vaya a trabajar.

Lei et ál. [LZP05] presentan un complejo sistema de posicionamiento para la comprobación de medidas en piezas de trabajo. Este sistema está compuesto por varias cintas transportadoras que mueven el objeto a examinar, además de por una mesa de posicionamiento mecánica que se encarga de situar el objeto en la posición requerida para su inspección.

Vom Stein [VS07] presenta un sistema industrial para la inspección de culatas de motores. Para poder capturar la información de toda la superficie de las culatas, se utiliza un brazo robótico. El autor justifica su uso, incidiendo en la flexibilidad que ofrece un brazo robótico, ya que con un cambio en el utillaje de agarre, el sistema se puede emplear para inspeccionar otros tipos de objetos con geometrías distintas.

Pernkopf [Per04] presenta un sistema para la inspección superficial de bloques de acero. El dispositivo que emplea en este caso para la automatización del sistema es básicamente una cinta transportadora. De esta forma consigue que toda la superficie del bloque de acero, pase a través del dispositivo de captura, en este caso, un sistema basado en triangulación láser.

Rahayem [Rah08], propone un sistema para la realización de ingeniería inversa sobre la información tridimensional de varios objetos. Para la automatización del proceso de captura de datos, emplea dos sistemas de automatización distintos. Por un lado, una mesa giratoria que permite poder inspeccionar gran parte de la superficie del objeto y, por otro, un brazo robótico, en cuyo extremo se ha adaptado un dispositivo de captura basado en triangulación láser. El brazo empleado en este caso posee 6 ejes y junto con la mesa giratoria, se pretende inspeccionar la mayor parte de la superficie del objeto posible.

En esta sección hemos visto cuatro ejemplos en los que se plantean distintas alternativas. Como ya hemos mencionado anteriormente, su elección dependerá de los requisitos del sistema final a construir. En algunos casos, esta automatización también puede venir forzada por el dispositivo de captura por el que se haya optado (véase la Sección 2.2.1). Por ejemplo, en el caso de los dispositivos basados en triangulación, se necesita movimiento para poder inspeccionar toda la superficie de un objeto. De forma similar, ocurre con los dispositivos que se basan en la obtención de la forma mediante el movimiento, en los que, como su propio nombre indica, se basan en secuencias de movimiento para la captura de la información tridimensional de la superficie.

2.2.3 Equipo de procesamiento

Este equipo es el encargado de transformar la información recogida por el dispositivo de captura en la salida del sistema de visión. En algunas ocasiones, sirve además para gestionar los distintos dispositivos que componen el sistema completo. Como ocurre con el resto de componentes, las características de este

equipo vendrán condicionadas por los requisitos del sistema final que se vaya a construir, esto incluye tanto el software como el hardware necesario.

Basándonos en los sistemas de visión artificial estudiados en la literatura, podemos destacar ciertas características comunes que tendrá que poseer el equipo o equipos a utilizar. La primera de ellas es la de poseer una alta capacidad de cómputo, ya que los dispositivos de captura de información tridimensional se caracterizan por obtener un gran volumen de datos. La segunda de ellas es la de poseer una alta potencia en cuanto a gráficos se refiere, ya que los sistemas trabajan con modelos tridimensionales de los objetos a inspeccionar, que deberán ser procesados por un hardware específico para así conseguir un funcionamiento fluido del sistema final.

2.3 Fases de un sistema de visión artificial

Malamas et ál. [MPZ⁺03] establecen una serie de características que deben aparecer en cualquier sistema de visión artificial aplicado a la inspección industrial. Estas son: la flexibilidad, la eficiencia en cuanto a rendimiento, la velocidad, el coste, la exactitud y la robustez.

Estas características no deben faltar a la hora de diseñar el sistema de visión, y para ello es importante definir las entradas y las salidas del mismo. El proceso intermedio entre estas entradas y salidas generalmente viene estructurado en base a la siguiente secuencia de pasos (véase la Figura 2.5): la adquisición de imágenes, que emplea los dispositivos vistos a lo largo Sección 2.2.1 para obtener los datos de entrada al sistema; el procesamiento de las imágenes, que se basa en la aplicación de distintos algoritmos sobre las imágenes recogidas en el primer paso para destacar las regiones de interés y disminuir el ruido ocasionado por los dispositivos de captura y de automatización; la selección de características, que se corresponde con la extracción de características de las regiones de interés que hayan sido marcadas en el paso anterior; y, por último, la clasificación de los datos dependiendo de la salida del sistema que haya sido planteada en la definición del mismo.

2.3.1 Adquisición de imágenes

En esta primera fase de adquisición de imágenes se va a comenzar obteniendo la información que va a servir como entrada al sistema de visión artificial. Para ello se emplearán diversos dispositivos de entrada ya estudiados en la Sección 2.2.1.



Figura 2.5: Fases de un sistema de visión artificial propuestas por Malamas et ál. [MPZ⁺03].

2.3.2 Procesado de imágenes

Una vez adquirida la información, se comienza con el tratamiento de la misma. Para ello, se elimina el ruido que hayan podido ocasionar los dispositivos de captura y automatización, mejorando la calidad de la imagen o imágenes capturadas mediante distintos algoritmos.

Como resultado de los tratamientos que se aplican en este punto, se obtienen las partes relevantes de las imágenes, también conocidas como regiones de interés, eliminando de esta forma las zonas de la imagen que no presentan relevancia para posteriores fases [PO02].

2.3.3 Extracción de características

La fase de extracción de características se basa en la obtención de información de las regiones de interés obtenidas en el paso anterior. Ejemplos de características extraídas de imágenes incluyen el tamaño, la posición, el contorno o medidas de textura.

Sobre las características extraídas se pueden aplicar distintos algoritmos con los que se puede realizar un filtrado previo, seleccionando únicamente las características relevantes de cara a la posterior clasificación.

2.3.4 Clasificación

En este último paso se clasifican las regiones de interés, en base a las características extraídas de cada una de ellas y empleando para ello distintas técnicas, bien estadísticas o basadas en aprendizaje automático. Como resultado final, se obtendrán la salida o salidas que hayan sido definidas para el sistema.

2.4 Sistemas de visión artificial para la inspección de calidad

Durante los últimos años se han ido planteando distintas soluciones basadas en visión artificial, que han ido dando respuesta a problemas concretos generalmente relacionadas con los controles de calidad.

Malamas et ál. [MPZ⁺03] plantean una clasificación en la que agrupan las diferentes soluciones que se han ido desarrollando durante los últimos años en base a su área de aplicación dentro del control de calidad. Estos grupos son los siguientes (véase la Figura 2.6): (i) la inspección de calidad dimensional, que es la encargada de la comprobación de medidas en los objetos a examinar; (ii) la inspección de calidad superficial, que busca defectos en la superficie de los objetos a examinar; (iii) la inspección de calidad estructural, que comprueba la presencia de todos los componentes de un objeto; y, por último, (iv) la inspección de calidad operacional, que comprueba que el funcionamiento de los objetos inspeccionados se ajusta a las especificaciones de fabricación.



Figura 2.6: Clasificación de aplicaciones de inspección de calidad propuesta por Malamas et ál. [MPZ⁺03].

2.4.1 Inspección de la calidad dimensional

Los sistemas que pertenecen a este grupo son aquéllos que se centran en la comprobación de las medidas del objeto a inspeccionar, determinando en cada caso si éstas se encuentran dentro de los parámetros establecidos. Además de esto, también se realizan comprobaciones de las características geométricas del objeto.

Estos sistemas se han ido aplicando a una gran variedad de industrias a lo largo de los últimos años. Un ejemplo de estas aplicaciones es la de la industria del embalaje. Tonshoff et ál. [TSI97] presentan un sistema que emplea visión artificial para la automatización del suministro de papel de máquinas de embalaje. El sistema permite localizar la bobina de papel seleccionada, para que el sistema la pueda coger y transportar hasta la máquina que la haya solicitado. En este mismo sentido, Kavoussanos y Pouliezios [KP00], proponen un sistema automático para la manipulación de palés y vaciado de sacos de polietileno. En este caso el sistema de visión artificial se encarga de comprobar la posición de los palés, así como la correcta disposición de los sacos de polietileno sobre los mismos.

Otra de las aplicaciones más extendidas en el control de la calidad dimensional, es el de la inspección y clasificación de las juntas de soldadura en las tarjetas de circuito impreso, también conocidas como PCB (Printed Circuit Board). Kim et ál. [KCMP99] presentan un sistema de visión artificial para la clasificación de defectos en estas uniones de soldadura. Para ello, escogieron un dispositivo de captura de obtención de forma mediando el sombreado. Con este dispositivo obtienen distintas imágenes de cada región de interés, extrayendo posteriormente características 2D y 3D. Finalmente, con estas características combinadas con una serie de clasificadores¹, consiguen construir un sistema válido en términos de velocidad y en tasas de reconocimiento.

Anand et ál. [AMSB99] presentan un sistema de visión artificial para el corte de materiales laminados. Este sistema, captura imágenes del material que se encuentra cortado de forma irregular y mediante el uso de redes neuronales², calcula el corte óptimo del mismo teniendo en cuenta las necesidades dimensionales requeridas, minimizando la cantidad de material desperdiciado. Un ejemplo de aplicación de este método es el de la industria textil, en la que es habitual encontrar material con cortes irregulares.

Jimenez et ál. [JJCP99] proponen un sistema automático basado en visión artificial para la localización de frutas esféricas en árboles en un entorno natural. Utilizan un dispositivo basado en triangulación láser, que obtiene imágenes con información de profundidad. Éstas posteriormente son procesadas para la localización de la fruta. Empleando este sistema los autores consiguen localizar el 100% de las piezas visibles y el 74% de las que se encuentran tapadas por la vegetación del árbol. Posteriormente, Jimenez et ál. [JCP00] integran el sistema de visión dentro un complejo sistema robótico, destinado a la recolección automática de fruta.

Otra aplicación típica que se puede englobar en este grupo es la de la medición de líneas de segmento. Shiau y Jiang [SJ99] proponen un algoritmo

¹Un clasificador es un algoritmo que implementa la clasificación, esto es, el problema de identificar a que categoría pertenece un nuevo elemento, basándonos en un conjunto de elementos como entrenamiento cuyas categorías son conocidas.

²Una red neuronal es un modelo matemático inspirado en las redes neuronales biológicas y formada por unidades sencillas interconectadas, denominadas neuronas artificiales.

heurístico para la evaluación de la eficiencia de un sistema de visión encargado de la medición de distintas propiedades de unos segmentos de línea. En este mismo sentido, Tsai [Tsa99] presenta un sistema para la inspección de del contorno de piezas circulares. Para ello emplea una transformada de Hough ¹ basada en dos pasos.

Magee y Seida [MS95] presentan un sistema de visión artificial basado en modelos para la estimación de la posición de objetos en entornos industriales en tiempo real. Torres et ál. [TSA⁺98] por su parte proponen un sistema para la inspección en tiempo real de impresiones de alta calidad. Los elementos inspeccionados en este caso se caracterizan por poseer diferencias mínimas entre ellos, por lo que el sistema debe poseer la mayor precisión posible y en tiempo real. Para ello el sistema emplea un algoritmo basado en operaciones morfológicas sobre las imágenes capturadas, que facilitan la posterior detección de posibles errores. Un buen ejemplo de aplicación de este sistema es la inspección de billetes.

Otra aplicación industrial perteneciente a esta categoría es la inspección de las roscas en tornillos para el cumplimiento de los estándares de fabricación. Hunsicker et ál. [HPL⁺94] presentan un sistema para este cometido, que emplea algoritmos de detección de bordes para localizar las regiones de interés en las imágenes. Posteriormente se empareja cada región con modelos almacenados y se miden las desviaciones existentes. En caso de que estas variaciones superen los máximos establecidos, se marcará la región como inválida. Este sistema ha conseguido sustituir a la inspección manual, debido principalmente a su alta tasa de acierto y a su funcionamiento en tiempo real. Hunter et ál. [HGT95], por su parte, presentan un sistema basado en modelos de forma activa. Este tipo de sistemas detecta puntos característicos de la forma de los objetos y almacena datos estadísticos de las relaciones existentes entre estos puntos. Un claro ejemplo de aplicación de este sistema es la de la inspección de los de distintos elementos presentes en los sistemas de frenado de los automóviles.

En este mismo sentido, Cuypers et ál. [CVGV⁺09] presentan un sistema para la medición de la calidad dimensional en objetos de grandes dimensiones. Concretamente, inspeccionan defectos en una pieza de fundición grande con un sistema de triangulación óptica basada en LED. Entre los defectos a inspeccionar se encuentran los que pueden aparecer en las superficies después del proceso de mecanizado, como es el caso de la falta de material en ciertas zonas.

¹La transformada de Hough es un algoritmo para el reconocimiento de patrones en imágenes que permite encontrar formas predeterminadas como líneas o círculos mediante un método estadístico.

2.4.2 Inspección de la calidad estructural

Las tareas más comunes presentes en este grupo, son principalmente la comprobación de componentes en piezas ensambladas y la detección de la presencia de objetos extraños adicionales. En esta sección veremos las distintas aplicaciones industriales que se engloban dentro de este grupo.

Uno de los usos más extendidos de este tipo de sistemas de inspección es el desarrollado para industrias dedicadas a la fabricación de componentes electrónicos. En este contexto Moganti et ál. [MEDT96] presentan un completo estado del arte en el que se proponen distintos algoritmos y estrategias para la inspección visual automatizada de placas de circuito impreso.

En esta misma línea, Cootes et ál. [CPJT96] plantean la validez de un nuevo método para la localización de distintos componentes en placas de circuito impreso. Su enfoque utiliza principalmente un modelo estadístico combinado con una búsqueda realizada en múltiples resoluciones. De esta forma, consiguen localizar con precisión los componentes y distinguir cuales de ellos son correctos y cuales no. Los resultados que presentan demuestran una tasa de error del 0,1 %, que junto con su capacidad de procesamiento en tiempo real, lo hacen completamente viable para la tarea en cuestión.

Otra aplicación de este tipo de sistemas se corresponde con la industria automovilística. En este sentido, Kim et ál. [KKS98] proponen un sistema para la detección de anomalías en el ensamblado de los componentes en una cadena de ensamblado de vehículos. Para ello, debe haber un operador humano que le indique al sistema las regiones que se van a inspeccionar, posteriormente, estas regiones se comparan con plantillas almacenadas por el sistema. Por último, los autores hacen referencia a que el trabajo presentado forma parte de un complejo sistema de inspección de toda una línea de ensamblado de vehículos.

La inspección de los componentes de las vías de tren es otra de las aplicaciones pertenecientes a este grupo. Velten et ál. [VKM99] exponen un sistema que combinan algoritmos de filtrado de imágenes para la detección de los elementos de sujeción de las vías, junto con redes neuronales para la comprobación de su correcta posición.

2.4.3 Inspección de la calidad operacional

La inspección de la calidad operacional se refiere a aquellas aplicaciones que se encargan de verificar el correcto funcionamiento de los productos inspeccionados, atendiendo para ello a los estándares de fabricación previamente definidos.

Un ejemplo claro de aplicación de este tipo de inspección es el de la soldadura láser. En este contexto Yeng et ál. [JML00], presentan un sistema de visión artificial que va examinando las superficies a soldar y determina el pun-

to donde se requiere realizar la soldadura, así como la cantidad apropiada de la misma para cubrir cada punto. Para ello, aplica algoritmos de eliminación de ruido sobre las imágenes y emplea filtros Laplacianos para la segmentación de las regiones existentes. Posteriormente, calcula la trayectoria y la cantidad de la soldadura en cada punto, y transmite esta información a un sistema automatizado de soldadura láser.

2.4.4 Inspección de la calidad superficial

Las áreas típicas de este grupo se centran principalmente en la inspección de la superficie de distintos elementos. Más concretamente se buscan posibles desperfectos que hayan podido surgir, como grietas o agujeros, así como una rugosidad excesiva o una textura inadecuada. Son varias industrias las que han ido incorporando a lo largo de los últimos años sistemas de visión artificial encargados de cumplir este cometido, entre ellas la textil, la maderera o la del metal. Estos sistemas generalmente han estado enfocados a la detección de defectos superficiales y a la verificación de la calidad superficial.

Bahlmann et ál. [BHR99] exponen un sistema basado en visión artificial para la inspección de calidad de costuras en superficies textiles. Para ello los autores emplean imágenes de bajo contraste en las que inspeccionan las ondulaciones existentes en los tejidos. A continuación, extraen características de las regiones de interés y clasifican cada una de ellas en base los requisitos establecidos.

Drake y Packianather [DP98] demuestran mediante un sistema basado en árboles de decisión ¹ en combinación con una topología de redes neuronales modular, una mayor eficiencia en la detección de defectos en láminas de madera. El diseño de la esta, depende de las variaciones en las características entre los elementos de las distintas clases.

Smith et ál. [SSH99] presentan un sistema para la inspección de superficies con un alto grado de reflectividad. La inspección realizada se compone de dos pasos principalmente, comenzando con una valoración de la superficie, y continuando con la clasificación de los defectos superficiales segmentados.

Tsai et ál. [TCC98] realizan un estudio empleando un sistema de visión artificial para la validación de la rugosidad de la superficie en piezas mecanizadas. Este sistema aplica la transformada discreta de Fourier ² en dos dimensiones y utiliza las características extraídas de la rugosidad como entrada a dos redes neuronales, siendo una de ellas para objetos que se encuentran posicionados de forma conocida, y la otra para objetos que se encuentran po-

¹Un árbol de decisión es un modelo de predicción basado en la construcción de diagramas de condiciones que ocurren de forma sucesiva.

²La transformada discreta de Fourier es un tipo de transformada discreta, que, transforma una función matemática en otra, obteniendo una representación de la misma en el dominio frecuencial.

sicionados de forma aleatoria.

Por su parte, Shafarenko et ál. [SPK97] presentan un nuevo método de segmentación para la inspección automática de granito. Este método de segmentación se basa en una combinación de características de color, así como de textura. Además emplea sobre la información de las imágenes a color la transformada de Watershed. Mediante este nuevo método, los autores consiguen unos buenos resultados no sólo para segmentar imágenes de granito, sino también para imágenes de color con ruido.

Otra aplicación perteneciente a este grupo es la de la inspección de la corrosión en tuberías de gas. Boyer y Ozguner [BO01] proponen una técnica para este cometido que compara las superficies a inspeccionar con modelos robustos, buscando desviaciones estadísticas, que implicarían la presencia de corrosión. El sistema es capaz de trabajar en tiempo real, pero ofrece unos resultados poco eficientes cuando se trata de zonas amplías de corrosión, provocado principalmente por el uso de una ventana deslizante para ir extrayendo las regiones de interés a examinar.

Xiang et ál. [XHY09] presentan el diseño de un sistema basado en visión artificial para la detección pequeños agujeros en la superficie del papel de aluminio. Para ello, y después de ajustar los niveles de contraste y de binarizar las imágenes, examinan los píxeles resultantes del proceso y realizan una validación de las porciones examinadas, comprobando que se ajustan a los estándares requeridos. Además de esto, demuestran que es capaz de realizar la inspección de la superficie del papel de aluminio en tiempo real de forma efectiva.

Existen numerosas aplicaciones industriales para los sistemas de visión artificial dedicadas a la inspección de la calidad superficial. Esto, implica que la gama de materiales distintos a inspeccionar sea muy amplia y que cada uno de ellos posea una textura superficial distinta. Estas texturas dificultan de una forma considerable la detección de defectos superficiales, ya que pueden inducir a errores a estos sistemas debido a las irregularidades que pueden presentar en la superficie.

2.4.5 Detección de defectos superficiales mediante el análisis de textura

La textura es una de las características que más información proporciona sobre una superficie. Basándose en esta afirmación, Xie [Xie08] propone una clasificación de los distintos enfoques que se han empleado para la detección de defectos superficiales mediante el análisis de la textura. Esta clasificación se compone de los siguientes cuatro enfoques (véase la Figura 2.7): los estadísticos, que miden la distribución espacial de los valores en cada uno de los píxeles; los estructurales, que emplean la disposición espacial de los elementos de la textura; los basados en filtro, que se basan en el uso de filtros de imagen;

y por último, los basados en modelo, que emplean modelos matemáticos para la detección de defectos.

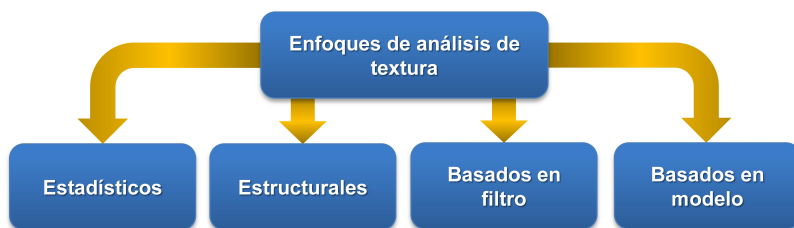


Figura 2.7: Clasificación de los enfoques de análisis de textura para la detección de defectos superficiales propuesta por Xie [Xie08].

2.4.5.1 Enfoques estadísticos

Los métodos de análisis estadístico miden la distribución espacial de los valores en cada uno de los píxeles. Son uno de los métodos más empleados en aplicaciones de visión artificial, debido principalmente a su sencillez. Dentro de ellos podemos diferenciar los siguientes cuatro tipos: las propiedades de histograma, la matriz de co-ocurrencia, el patrón binario local, y la autocorrelación.

- **Propiedades de histograma.** El histograma de una imagen es una representación que muestra en forma de diagrama de barras el número de píxeles que existen para cada una de las intensidades en la imagen. Por ejemplo, para una imagen en escala de grises hay 256 intensidades posibles, ya que sólo existe un canal, mientras que para una imagen en color codificada en RGB¹ encontramos 256 intensidades posibles para cada una de las componentes de color. En la Figura 2.8 podemos ver un ejemplo de un histograma simplificado. Como se puede observar el eje horizontal presenta los valores posibles para una determinada variable, mientras que en el eje vertical encontramos el número de casos en los que aparece cada uno de los valores, obteniendo de esta forma una impresión visual sobre la distribución de los datos.

Las propiedades que se extraen de él se pueden clasificar en dos tipos dependiendo de su complejidad. En el primero de ellos encontramos estadísticas tales como la media, la desviación estándar, la mediana o la varianza. En el segundo grupo están estadísticas de comparación entre histogramas, como por ejemplo la distancia Bhattacharyya², la inter-

¹El modelo RGB (Red, Green, Blue) de un color hace referencia a la composición de ese color en términos de intensidad de los colores primarios: rojo, verde y azul).

²La distancia de Bhattacharyya mide la similitud entre dos distribuciones de probabilidad continuas o discretas.

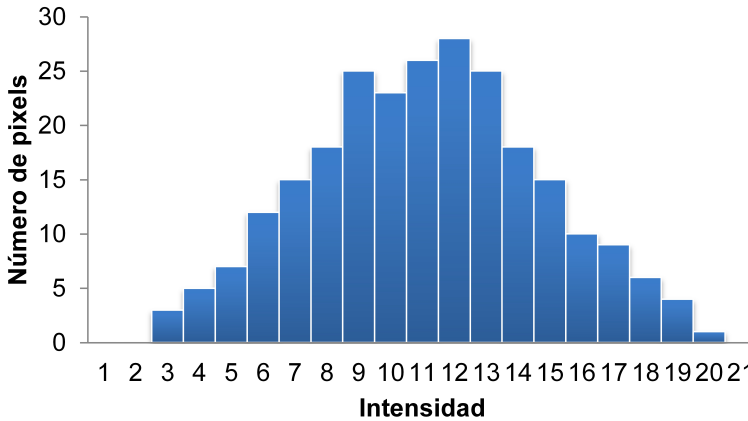


Figura 2.8: Ejemplo simplificado de un histograma.

sección entre histogramas o el coeficiente de correlación normalizado¹. Las características principales que presentan para la detección de defectos superficiales son su eficacia, simplicidad y bajo coste [PMV02]. Una forma de mejorar la eficacia de estos métodos es la de emplear regiones de menor tamaño, empleando para ello técnicas de segmentación [XMT06, BSPC05].

- **Matriz de co-ocurrencia.** La matriz de co-ocurrencia de escala de grises también conocidas como GLCM [HSD73] son una de las características más utilizadas. A grandes rasgos es una estadística de segundo orden que representa la dependencia espacial entre dos niveles de gris mediante una matriz bidimensional y un vector de desplazamiento dado. Matemáticamente, la matriz de co-ocurrencia MC se define como:

$$MC_{\Delta x, \Delta y}(i, j) = \sum_{p=1}^n \sum_{q=1}^m \begin{cases} 1, & \text{Si } I(p, q) = i \text{ e } I(p + \Delta x, q + \Delta y) = j \\ 0, & \text{en caso contrario} \end{cases} \quad (2.1)$$

Siendo I la imagen de tamaño $n \times m$ sobre la que se basa la matriz y Δx y Δy dos parámetros para indicar el desplazamiento. A partir de ella se pueden extraer distintas características como la energía, la entropía, el contraste, la homogeneidad o la correlación, con las que se pueden detectar defectos en superficies con textura [IRV96, BWBK97, Mon04].

¹El coeficiente de correlación es una medida estadística que proporciona información sobre la relación lineal entre dos variables.

Por otra parte existen factores que influyen de forma negativa en la efectividad de la matriz de co-ocurrencia destacan: la inexistencia de una metodología aceptada para determinar los valores del vector de desplazamiento, y la dimensión de la propia matriz, ya que cuantas más entradas presente menor será la fiabilidad de la información que representa [Mon04].

- **Patrón binario local.** El patrón binario local, también conocido como LBP fue introducido por primera vez por Ojala et ál. [OPH96], como una característica de medida complementaria e invariante al contraste de una imagen. Para aplicar el patrón binario local primero dividimos la imagen en regiones de 3 x 3 pixels, posteriormente se emplea la siguiente fórmula sobre cada una de las regiones:

$$PBL(x_c, y_c) = \sum_{n=1}^7 s(i_n - i_c) 2^n \quad (2.2)$$

Donde i_c es el valor del pixel central con coordenadas (x_c, y_c) , i_n se corresponde con el valor de los ocho píxeles del alrededor y la función $s(x)$ se define como:

$$s(x) = \begin{cases} 1, & \text{Si } x \geq 0 \\ 0, & \text{Si } x < 0 \end{cases} \quad (2.3)$$

Las numerosas aplicaciones que se han realizado del patrón binario local para la inspección de superficies, lo sitúan como uno de los métodos más robustos existentes en la literatura [OPM02, MTP03, Mon04, MP05], principalmente por que su valor no se ve afectado por rotaciones en la iluminación o en la propia textura.

- **Autocorrelación.** Esta característica se basa en la repetitividad de algunas texturas de la naturaleza, como ocurre por ejemplo con el caso de los textiles. Concretamente mide la correlación entre un imagen y la translación de ella misma en base a un vector de desplazamiento. En el caso de las texturas regulares y repetitivas la medida de autocorrelación se vuelve irregular. Los principales inconvenientes que presenta esta característica de segundo orden son principalmente su alta sensibilidad al ruido, así como su baja eficacia a la hora de examinar texturas aleatorias. Aun así se en algunos ejemplos concretos de inspección de superficies se consiguen buenos resultados [Woo90, HC04].

2.4.5.2 Enfoques estructurales

Estos enfoques tienen como premisa que una textura puede ser identificada mediante una serie de primitivas o elementos, junto con la disposición espacial

que estos presentan [VNP86], por lo que la primera tarea a realizar es la de extraer estas primitivas, generando un modelo que represente la disposición de cada una de ellas. El tamaño y la forma de estas primitivas puede variar de forma considerable dependiendo de la textura, pudiendo ser píxeles individuales, regiones que presenten un mismo nivel de gris o incluso segmentos de una línea. En cuanto al cálculo del modelo de representación de la disposición, se pueden emplear las relaciones geométricas entre las primitivas extraídas, o bien mediante el aprendizaje de las propiedades estadísticas de las mismas.

Chen y Jain [CJ88] proponen un método para la identificación de defectos superficiales en imágenes obtenidas de textiles. Para ello, comienzan binarizando¹ las imágenes mediante un análisis de histograma. Posteriormente, crean una estructura de datos en la que almacenan la información de la textura. Finalmente, obtienen distintas medidas estadísticas de los datos a inspeccionar y los comparan con esas mismas medidas de superficies libres de defecto y almacenadas previamente.

Kittler et ál. [KMM⁺94] por su parte, emplean el algoritmo de agrupamiento k-means² para dividir de forma aleatoria la imagen en regiones que presenten cierta homogeneidad. Posteriormente, analizan estas regiones para localizar las primitivas que representen la textura de esas zonas, extrayendo para ello características como el tamaño, el perímetro o la distribución espacial.

Otro método que se engloba dentro de los enfoques estructurales es el presentado por Mallik-Goswami y Datta [MGD00], que se basa en la aplicación de distintos operadores morfológicos³ con los que consiguen resaltar los defectos existentes en imágenes tomadas de tejidos.

Wen y Xia [WX99] por su parte proponen un método que comienza con la detección de bordes en imágenes de piezas de cuero. Posteriormente extraen los segmentos de borde detectados en el paso previo y evalúan mediante métodos estadísticos la validez de cada uno de estos segmentos.

2.4.5.3 Enfoques basados en filtro

Este tipo de enfoques se basan en la aplicación de un conjunto de filtros predefinidos sobre las imágenes a examinar, extrayendo posteriormente la información de la respuesta de la imagen al filtro aplicado. Los enfoques basados en filtros se pueden dividir en los siguientes tres tipos: el filtrado del domi-

¹Una imagen binarizada es aquella cuyos píxeles sólo pueden tener dos valores posibles, blanco o negro.

²El algoritmo de agrupamiento k-means es un método que tiene como objetivo la división de n observaciones en k grupos, basándose para ello en la cercanía de cada observación con respecto al valor medio de cada grupo.

³Un operador morfológico es operación que emplea técnica para el análisis y tratamiento de estructuras geométricas basada en teoría de conjuntos, topología, teoría de retículos y funciones aleatorias.

nio espacial, el análisis del dominio frecuencial y por último el filtrado de la frecuencia espacial.

- **Filtrado del dominio espacial.** Los métodos de filtrado en el dominio espacial se caracterizan por emplear filtros de gradiente para la extracción de bordes o puntos significativos de la imagen. Dentro de este tipo de filtros podemos encontrar el filtro Sobel, el de Robert, el Canny o el Laplaciano entre otros. Unos de los más empleados para la detección de defectos en superficies texturadas son los Eigenfilters[Ade83, MMT04], que se componen de un conjunto de máscaras obtenidas a través de la transformada de Karhunen-Lóeve¹.
- **Análisis del dominio frecuencial.** Uno de los filtros más característicos del dominio frecuencial es la transformada de Fourier². Los métodos que usan este tipo de filtros para la inspección de superficies demuestran buenos resultados cuando la textura a examinar es repetitiva y regular, mientras que por el contrario, si presenta un cierto grado de aleatoriedad se hacen del todo inviables [TH99, CP00, TH03].
- **Filtrado de la frecuencia espacial.** La frecuencia espacial es una característica común a cualquier estructura periódica en el espacio, más concretamente, es una medida de la frecuencia de repetición de los componentes sinusoidales de la estructura por una unidad de distancia dada [DVAT82].

En cuanto a los tipos de filtros empleados para clasificación e inspección de superficies, son destacables los filtros Gabor [KP02, BBL02, BB03, Mon04], así como los filtros Wavelets [Sch05, YPY05, TL08]

2.4.5.4 Enfoques basados en modelo

Los enfoques que se engloban en este grupo son aquéllos que se basan en modelos matemáticos para determinar la validez de una superficie. Dentro de ellos encontramos tres tipos distintos: los modelos fractales, los modelos de campos aleatorios de Markov; y los modelos Texem.

- **Modelos fractales.** Los modelos fractales, propuestos por Mandelbrot [Man83], son primitivas geométricas irregulares y similares entre si que están presentes en la naturaleza. Algo que define de una forma clara este tipo de modelos es que los fragmentos de un objeto fractal son copias estadísticas del mismo objeto entero. En la literatura encontramos

¹Es una transformada basada en el análisis de componentes principales, con cuyo resultado se busca concentrar la energía de una imagen de la forma más eficiente posible.

²La transformada de Fourier transforma una función matemática en otra, obteniendo una representación de esta en el dominio frecuencial

distintos trabajos que usan modelos fractales para la inspección de superficies [CP98, CP02]. Estos ejemplos destacan el limitado ámbito de aplicación de los modelos fractales, ya que no todas las texturas pueden ser representadas mediante estas primitivas.

- **Modelos de campos aleatorios de Markov.** Los modelos de campos aleatorios de Markov se utilizan como un método para la caracterización de las influencias mutuas entre los diferentes píxeles de una imagen. Existen varios ejemplos de métodos que emplean este tipo de modelos para la inspección superficial de forma eficaz [BAEG00, Li09]
- **Modelos Texem.** Los modelos Texem, son un método propuesto por Xie y Mirmehdi [XM05a, XM05b, XM07] para el análisis de texturas aleatorias. Parten de la base de que la imagen obtenida de la textura está compuesta por dos capas, siendo la primera de ellas un superposición de ejemplares de la textura, pudiendo estar solapados sobre la segunda capa. Cada uno de estos ejemplares de textura se caracteriza por unos valores medios y sus correspondientes covarianzas.

2.5 Sumario

A lo largo de este capítulo, hemos realizado un estudio completo de los sistemas de visión artificial presentes en la literatura aplicados a la inspección de calidad en procesos productivos. Una de las conclusiones que podemos extraer de la amplia diversidad de métodos, algoritmos y dispositivos existentes, es que no existe un sistema capaz de hacer frente a las necesidades de cada caso en concreto [PO03], debido principalmente a la complejidad que presenta, por lo que su diseño va a depender de las características de los objetos a examinar, de los defectos a inspeccionar, así como del entorno en el que se vaya a integrar.

En cuanto a los dispositivos que componen el sistema, no existe ninguna configuración que destaque sobre todas las demás. Esto ocurre tanto con los dispositivos de captura como con los automatización. Como hemos podido observar cada uno tiene sus fortalezas y sus debilidades, por lo que su elección vendrá determinada por los requisitos de funcionamiento que se hayan establecido para el sistema final. Una de las cosas a tener en cuenta una vez escogidos los dispositivos es el formato en el que nos presenta la información que captura, ya esta representación determinará todos los procesos posteriores que realicemos.

Una vez que hayamos realizado la especificación de requisitos, comenzaremos con el diseño global del sistema, para lo que será necesario determinar los componentes que mejor se adapten a las necesidades planteadas. Posteriormente, podremos comenzar con la aplicación de los algoritmos y estrategias necesarias para la detección y clasificación de los posibles defectos existentes.

Finalmente, queremos destacar que ninguna de las soluciones estudiadas cumple con los requisitos de funcionamiento que hemos definido en nuestro caso, por lo que necesitaremos crear un nuevo sistema que permita desarrollar distintos métodos para la detección y clasificación de defectos superficiales en piezas de fundición de hierro nodular.

«La vitalidad se revela no solamente en la capacidad de persistir sino en la de volver a empezar.»

Francis Scott Fitzgerald (1896-1940)

CAPÍTULO

3

Arquitectura del sistema de visión artificial propuesto

PARA realizar la captura de la información superficial de las piezas de fundición de hierro nodular, hemos diseñado, desarrollado e implantado un sistema de visión artificial. Con él, buscamos cubrir los requisitos planteados para la inspección de calidad superficial. Tal y como hemos podido comprobar en el capítulo anterior (véase la Sección 2.2.1), la elección del dispositivo de captura condiciona no sólo el formato en el que se captura la información, sino que también influye en el diseño y elección del resto de componentes.

A lo largo de este capítulo, vamos a exponer el diseño y el funcionamiento básico del sistema de visión artificial que proponemos para la captura de información, así como los primeros pasos de preprocesado que aplicamos a los datos capturados por el mismo.

El resto del capítulo continúa de la siguiente forma. La Sección 3.1 expone de un modo detallado los dispositivos que empleamos para la construcción del sistema de visión artificial. La Sección 3.2 especifica la composición del conjunto de datos real que se ha empleado para el estudio propuesto. La Sección 3.3 describe el proceso que aplicamos a la información que vamos capturando mediante el sistema de visión artificial construido. La Sección 3.4 expone el proceso que se aplica a la información obtenida del sistema de visión artificial

para normalizarla. La Sección 3.5 describe las diferentes representaciones que va a tener la información capturada para posteriores procesos. Por último, la Sección 3.7 resume las aportaciones realizadas en este capítulo.

3.1 Elementos del sistema de visión artificial propuesto

Una vez realizada la recopilación de requisitos que debe cumplir el sistema, hemos comenzado el diseño del mismo, eligiendo en cada caso los dispositivos que lo van a componer. Básicamente, nos vamos a centrar en los tres dispositivos necesarios que forman el sistema final: (i) el dispositivo de captura, que como ya hemos analizado anteriormente, es el encargado de registrar la información de entrada al sistema; (ii) el dispositivo de automatización, cuya misión es automatizar el proceso de captura; y, por último, (iii) el equipo de procesamiento, que en este caso se va a encargar no sólo de procesar la información que vaya recogiendo el dispositivo de captura, sino también de la gestión de los dispositivos que componen el sistema.

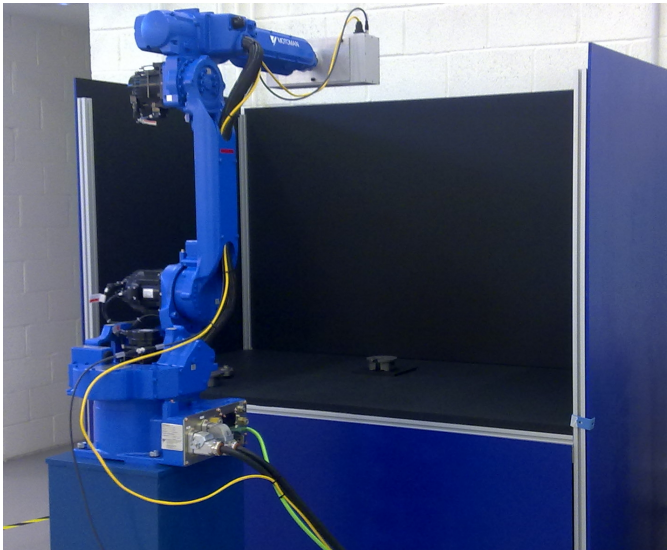


Figura 3.1: Prototipo del sistema de visión artificial propuesto.

3.1.1 Dispositivo de captura

Para capturar la información superficial, hemos optado por un dispositivo basado en triangulación láser (véase la Sección 2.2.1.2), en concreto, es una

Sick|IVP Ruler E600 SB. Las razones por las que hemos decidido emplear este tipo de dispositivo son principalmente su alta precisión y su gran capacidad de adaptación a entornos adversos. Tenemos que tener en cuenta que el sistema debe ser capaz de trabajar en una fundición de hierro nodular, que se caracteriza principalmente por unas altas temperaturas, unas condiciones de iluminación no controladas y una gran cantidad de polvo en el ambiente. Otra de las ventajas que presenta el dispositivo es la posibilidad de elegir la potencia del láser empleado. En nuestro caso, y debido principalmente a que la superficie de las piezas examinar es muy oscura, hemos optado por un láser potente de la clase 3B. Además de todo esto, tanto el proyector láser como el sensor óptico empleado para la captura, se encuentran calibrados y ensamblados en una caja de aluminio estando protegidos, de esta forma, de las adversidades de la planta de fundición.

3.1.2 Dispositivo de automatización

Para la automatización del proceso, hemos recurrido a un brazo robótico. Ésta únicamente, la aplicamos al proceso de captura y no al de posicionamiento de la pieza. Este último se realiza de forma manual ya que implica una gran complejidad, debido principalmente, a las diferencias geométricas entre los distintos modelos de piezas con las que se va a trabajar. En concreto, el modelo de brazo robótico empleado es un Motoman NH6-10. Este dispositivo posee cinco ejes, lo que lo convierte en una herramienta muy versátil de cara a la inspección de piezas independientemente del tamaño que tengan. Además, el robot es capaz de soportar el peso de la cámara y del soporte de aluminio que los une, que ronda los 8 kilogramos en total.

Otra de las decisiones que hemos tomado, es la de ensamblar la cámara al extremo del brazo, ya que de esta forma el sistema se puede adaptar a modelos de pieza de tamaños heterogéneos. Para realizar este ensamblado, hemos fabricado un utillaje de aluminio, minimizando las vibraciones que pudieran influir en la datos capturados como consecuencia de los movimientos del brazo robótico.

En cuanto al posicionamiento de la pieza, hemos empleado moldes de silicona que se encajan en una mesa de trabajo. Este método presenta una gran sencillez pero implica la manipulación de un operario, por lo que no sería viable su implantación en un entorno real, siendo necesario recurrir a cualquiera de las soluciones propuestas vistas en la Sección 2.2.2.

3.1.3 Equipo de procesamiento

Como equipo de procesamiento, hemos empleado una estación de trabajo compuesta por un procesador Intel XENON E5506, una memoria RAM de 6 gigabytes DDR3 y una tarjeta gráfica Nvidia QUADRO FX1800. Con estos com-

ponentes, hemos buscado cubrir las necesidades del sistema de visión artificial en cuanto a la gestión de los distintos dispositivos que lo componen, así como las del procesamiento de toda la información necesaria para detectar y clasificar defectos superficiales.

3.2 Conjunto de datos empleado para la validación

Para comprobar la hipótesis planteada, hemos trabajado con un conjunto de datos real para realizar los experimentos necesarios. Este conjunto está formado por 639 piezas de un modelo perteneciente al sector de la automoción.

Dentro del conjunto de piezas que hemos empleado, existen piezas correctas y con defecto. Del primer grupo tenemos 236 ejemplares, mientras que por otra parte, disponemos de 403 piezas que presentan uno o más defectos superficiales. En la Tabla 3.1 observamos el número de defectos que encontramos de cada tipo.

Tabla 3.1: Número de apariciones de cada tipo de defecto.

Tipo	Número de apariciones
<i>Inclusión</i>	392
<i>Unión fría</i>	8
<i>Falta de llenado</i>	14

Como podemos comprobar en la Tabla 3.1, el número de apariciones de cada uno de los tipos de defecto varía de forma considerable, debido principalmente a que, en este caso concreto, las inclusiones son uno de los defectos superficiales que suelen aparecer más comúnmente. Por el contrario, las uniones frías son un defecto que suele ocurrir en un número reducido de ocasiones, de la misma forma que ocurre en el caso de las faltas de llenado.

3.3 Obtención de los datos

El dispositivo de captura que hemos escogido nos ofrece la posibilidad de trabajar de dos formas distintas: la primera de ellas se denomina *modo imagen*, y como su propio nombre indica, obtiene una imagen de la escena en blanco y negro; la segunda, por el contrario, denominada *modo medición*, nos devuelve un vector con las alturas de cada uno de los puntos de la porción de superficie proyectada por el láser.

Para nuestro caso concreto, hemos optado por emplear el *modo medición*, ya que con él obtenemos información tridimensional de la superficie. Esta decisión implica un mayor tiempo para la adquisición de los datos, ya que

en cada captura se recogen únicamente los datos de la superficie iluminada por la línea láser, cuyo grosor es de 0,2 milímetros.

En este sentido, para poder recoger la información relativa a toda la superficie, el brazo robótico realiza un movimiento lineal, permitiendo que la cámara realice una captura cada 0.2 mm. De esta forma, iremos recogiendo los vectores que estarán compuestos por las diferentes alturas de forma secuencial. A partir de este punto, denominaremos a cada uno de esos vectores con el nombre de *perfil*. Por lo que formalmente podemos afirmar que una superficie escaneada $\mathcal{S}$ está formada por un conjunto de perfiles $\mathcal{P}$ siendo $\mathcal{S} = \{\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_{n-1}, \mathcal{P}_n\}$.

Cada uno de estos perfiles son vectores $\vec{p}$ compuestos por las alturas en cada uno de los puntos $p_{x,y}$. Uniéndolos, generamos la matriz de alturas $\mathcal{A}_{m \times n}$, que representa la superficie total escaneada $\mathcal{S}$ y que representamos de la siguiente forma.

$$\mathcal{A}_{m \times n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \\ & & \dots & & \\ a_{m-1,1} & a_{m-1,2} & \dots & a_{m-1,n-1} & a_{m-1,n} \\ a_{m,1} & a_{m,2} & \dots & a_{m,n-1} & a_{m,n} \end{pmatrix} \quad (3.1)$$

donde cada $a_{x,y}$ representa la altura en cada uno de los puntos del espacio (x, y) . Las dimensiones m y n vendrán determinadas por el tamaño de la superficie inspeccionada.

Una vez formada la matriz de alturas $\mathcal{A}_{m \times n}$, comenzamos con su procesamiento. Así, en un primer lugar se eliminan los puntos que pertenecen a la mesa de trabajo, a la que denominaremos $\mathcal{T}_{o \times p}$ siendo $0 < o \leq m$ y $0 < p \leq n$. Para ello, establecemos un umbral σ que vendrá determinado por la máxima altura de los puntos que pertenecen a la mesa de trabajo o, dicho de otro modo:

$$\sigma = \operatorname{argmax}_{t_{x,y} \in \mathcal{T}_{o \times p}} t_{x,y} \quad (3.2)$$

determinando x e y la posición del elemento a en la matriz $\mathcal{T}_{o \times p}$.

Una vez calculado el umbral σ , se lo restamos a cada punto de la matriz, estableciendo como altura mínima el valor 0, tal que:

$$\forall a_{x,y} \in \mathcal{A}_{m \times n} : ((a_{x,y} \triangleq a_{x,y} - \sigma) \mid a_{x,y} \geq 0) \quad (3.3)$$

Como resultado de este proceso, hemos descartado la información que no presenta ninguna utilidad, como por ejemplo, los datos de la mesa de trabajo. A continuación, eliminamos la mayor cantidad de esta información para que la matriz resultante sea más fácil de manipular. Así, creamos una nueva matriz

$\mathcal{B}_{q \times r}$ (véase la Ecuación 3.4), donde q es el número de filas que tienen todos sus elementos distintos de cero en la matriz $\mathcal{A}_{m \times n}$ y r el número de columnas que tienen todos sus elementos distintos de cero en dicha matriz, por lo cual se cumple que $0 < q \leq m$ y $0 < r \leq n$.

$$\mathcal{B}_{q \times r} = \begin{pmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,r-1} & b_{1,r} \\ b_{2,1} & b_{2,2} & \dots & b_{2,r-1} & b_{2,r} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{q-1,1} & b_{q-1,2} & \dots & b_{q-1,r-1} & b_{q-1,r} \\ b_{q,1} & b_{q,2} & \dots & b_{q,r-1} & b_{q,r} \end{pmatrix} \quad (3.4)$$

3.4 Normalización de los datos

En este punto, tenemos los datos de las superficies representados en matrices de alturas, a las que les hemos aplicado distintos algoritmos para eliminar la información que no es relevante. Debido a ciertos aspectos de la problemática que estamos abordando, estas matrices $\mathcal{B}_{q \times r}$ que hemos generado hasta ahora puede que no tengan las mismas dimensiones. Para poder comparar en fases posteriores dos matrices entre sí, es necesario que éstas tengan las mismas dimensiones.

Para ello, comenzamos con un conjunto $\mathcal{C}$ formado por ℓ matrices $\mathcal{B}_{q_i, r_i}$ donde q_i y r_i son el alto y el ancho de la matriz i -ésima del conjunto de matrices $\mathcal{C}$. A fin de normalizar las matrices del conjunto $\mathcal{C}$, comenzamos determinando las dimensiones que van a tener. Éstas se corresponden con la altura y la anchura máxima, q' y r' del conjunto de matrices de entrada $\mathcal{C}$, tal que:

$$q' = \operatorname{argmax}_{q_i} \mathcal{B}_{q_i \times r_i} \in \mathcal{C} \quad (3.5)$$

donde q' es la máxima altura de todas las matrices $\mathcal{B}_{q_i \times r_i}$ dentro del conjunto de matrices $\mathcal{C}$. La anchura máxima se calcula de forma similar:

$$r' = \operatorname{argmax}_{r_i} \mathcal{B}_{q_i \times r_i} \in \mathcal{C} \quad (3.6)$$

Posteriormente, redimensionamos cada una de estas matrices con el nuevo ancho q' y el nuevo alto r' , asignando los nuevos valores:

$$\forall \mathcal{B}_{q_i \times r_i} \in \mathcal{C} : \mathcal{B}'_{q' \times r'} \triangleq \text{alinear}(\mathcal{B}_{q_i \times r_i}, q', r') \quad (3.7)$$

donde *alinear* es una función que mueve los valores de la matriz original $\mathcal{B}_{q_i \times r_i}$ a la nueva matriz normalizada en altura q' y en anchura r' denotada por $\mathcal{B}'_{q' \times r'}$.

$$alinear(\mathcal{B}_{q \times r}, q', r') = \mathcal{B}'_{q' \times r'} = \begin{pmatrix} b'_{1,1} & b'_{1,2} & \cdots & b'_{1,r'-1} & b'_{1,r'} \\ b'_{2,1} & b'_{2,2} & \cdots & b'_{2,r'-1} & b'_{2,r'} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ b'_{q'-1,1} & b'_{q'-1,2} & \cdots & b'_{q'-1,r'-1} & b'_{q'-1,r'} \\ b'_{q',1} & b'_{q',2} & \cdots & b'_{q',r'-1} & b'_{q',r'} \end{pmatrix} \quad (3.8)$$

donde en cada elemento de la nueva matriz $b'_{i,j} \in B'_{q' \times r'}$ almacenaremos los elementos de la matriz original $\mathcal{B}_{q \times r}$, desplazando su nueva posición en la nueva matriz $B'_{q' \times r'}$, tal que:

$$\forall b_{i,j} \in \mathcal{B}_{q \times r} : b'_{i+(q'/2)-(q/2), j+(r'/2)-(r/2)} \in \mathcal{B}'_{q' \times r'} \triangleq b_{i,j} \quad (3.9)$$

alineando todas las matrices y normalizando, de este modo, sus diferentes dimensiones, formando un conjunto de matrices normalizadas $\mathcal{C}'$ compuesto con las matrices normalizadas.

3.5 Representaciones utilizadas

Una vez normalizadas las matrices, comenzamos a generar las diferentes representaciones de la información capturada. A lo largo de esta sección, vamos conocer las representaciones que vamos a emplear y como las generamos.

Las cuatro representaciones generadas en este punto son: (i) la matriz de alturas, que como su nombre indica es una matriz que representa el valor de la altura de la superficie en cada punto; (ii) el mapa de alturas en escala de grises, que representa la información de la matriz de alturas de forma gráfica, y más concretamente en escala de grises; (iii) la matriz de vectores normales, que muestra la información de la dirección de la superficie en cada punto representada en forma de vector tridimensional; y, por ultimo, (iv) el mapa de vectores normales codificado en RGB, que representa la información de la matriz de normales de forma gráfica codificada en los tres canales del modelo RGB.

3.5.1 Matriz de alturas

La matriz de alturas es la representación con la que hemos estado trabajando hasta este punto (véase la Ecuación 3.1). Los valores que contiene, nos sirven para poder reconstruir un modelo tridimensional de la superficie escaneada.

En la Figura 3.2, podemos observar una matriz de alturas preprocesada y renderizada con una malla 3D. Para ello, hemos empleado una malla o rejilla en tres dimensiones, donde se representa la altura de la matriz en cada punto.

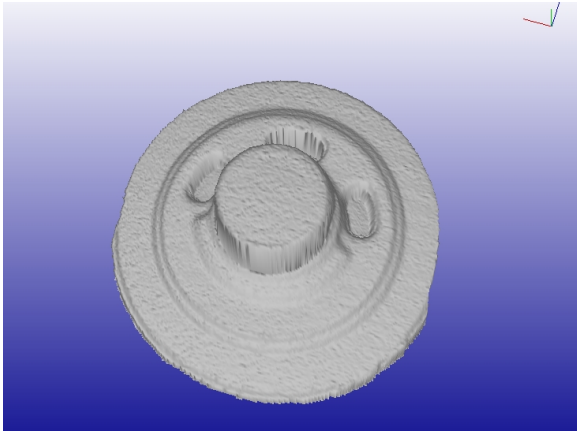


Figura 3.2: Visualización de la matriz de alturas mediante el renderizado de una malla 3D.

3.5.2 Mapa de alturas en escala de grises

El mapa de alturas en escala de grises es una representación gráfica de la matriz de alturas. Para generarlo, lo primero que hacemos es normalizar las alturas de la matriz en base a los distintos valores de gris posibles, que hacen un total de 256, correspondiendo el valor 0 con el color negro y el valor 255 con el color blanco (véase la Figura 3.3).

Para obtener el mapa de alturas, comenzamos transformando cada uno de los valores de la matriz de alturas a su valor correspondiente en un rango, cuyo valor mínimo es 0 y el máximo es 255. La altura resultante en cada punto será el valor que le asignaremos a los canales RGB de la imagen, ya que una imagen en escala de grises se caracteriza por tener el mismo valor para todos sus canales.

3.5.3 Matriz de vectores normales

En un espacio tridimensional podemos definir el vector normal $\vec{n}$ a una superficie $\mathcal{S}$ en un punto $\mathcal{P}$, como aquel vector que es perpendicular al plano tangente $\mathcal{T}$ a la superficie $\mathcal{S}$ y que contiene al punto $\mathcal{P}$.

En informática gráfica, un vector normal es aquél que determina la orientación de una superficie. Este vector es un elemento esencial a la hora de optimizar el proceso de dibujo de superficies, ya que contiene una gran cantidad de información sobre la superficie.

En la Figura 3.4 podemos observar en la primera de las imágenes los vectores normales para cada uno de los triángulos de la superficie. Posteriormente, para calcular el vector normal en cada uno de los puntos de la matriz de



Figura 3.3: Mapa de alturas en escala de grises de la superficie escaneada.

alturas, promediamos los valores de los vectores normales de los triángulos adyacentes a cada punto.

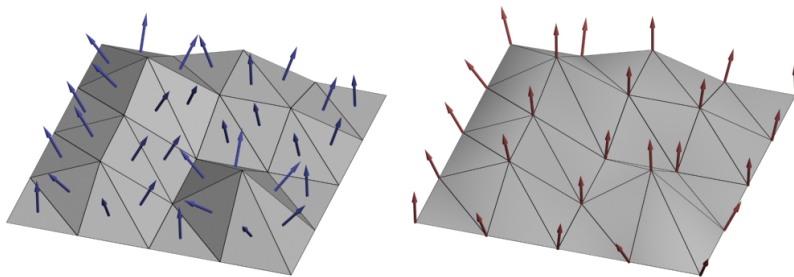


Figura 3.4: Visualización de la matriz de vectores normales sobre una malla 3D.

De esta representación podemos extraer tres matrices, una por cada uno de las componentes del vector normal en cada punto, $\mathcal{X}_{m \times n}$ que se corresponde con los valores para la componente x , $\mathcal{Y}_{o \times p}$ que se corresponde con los valores para la componente y , y $\mathcal{Z}_{q \times r}$ que se corresponde con los valores para la componente z , siendo $m = o = q$, y $n = p = r$.

3.5.4 Mapa de vectores normales codificado en RGB

De forma similar a la creación del mapa de alturas en escala de grises, también hemos generado una imagen para representar la matriz de vectores normales

de forma gráfica. Esta imagen, representa los distintos componentes de los vectores normales mediante los canales RGB de la imagen. En la Figura 3.5 podemos ver un ejemplo del mapa generado.

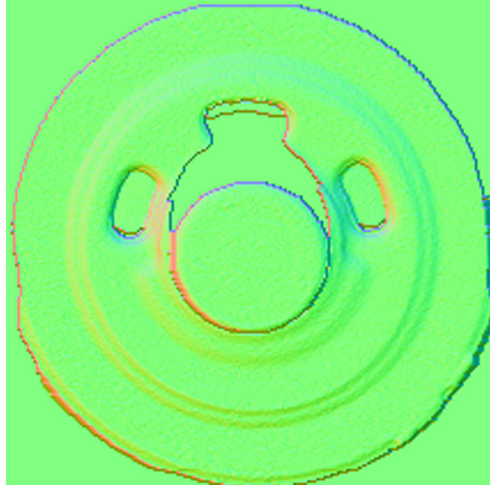


Figura 3.5: Mapa de vectores normales codificados en RGB de la pieza escaneada.

Para obtener el mapa de vectores normales codificado en RGB, comenzamos transformando cada uno de los valores de cada una de las tres matrices a su valor correspondiente en un rango cuyo valor mínimo es 0, y el máximo es 255. Posteriormente, creamos una imagen codificada en RGB con las mismas dimensiones que las tres matrices, en la que asignamos cada uno de los puntos de la matriz $\mathcal{X}_{m \times n}$ al componente R de la imagen, los puntos de la matriz $\mathcal{Y}_{o \times p}$ a la componente G y los puntos de la matriz $\mathcal{Z}_{q \times r}$ a la componente B.

3.6 Segmentación de las imágenes

Podemos definir la segmentación de una imagen, como el proceso de subdividir la imagen en regiones inconexas [Cas96]. Estas regiones que se detectan, se caracterizan por pertenecer a zonas en las que hay algún defecto, así como estructuras regulares propias de la superficie de la pieza [MFJ03]. Existe una gran variedad de métodos de segmentación, adaptado cada uno de ellos a las características del entorno de trabajo particular. A continuación, describimos un nuevo proceso para segmentar las piezas basado en el mapa de vectores normales codificado en RGB.

3.6.1 Proceso de segmentación

El proceso de segmentación que planteamos, emplea las piezas que no contienen ningún defecto como modelos contra los que comparar las piezas rechazadas. Además de estos modelos, empleamos distintos filtros de imagen para eliminar el ruido de algunos casos y para mejorar la extracción de información en otros.

En función del resultado de la pieza, se siguen subprocesos distintos. En caso de que la pieza sea correcta, eliminamos el ruido que pueda contener. Por contra, si la pieza contiene algún defecto superficial, empleamos un proceso que elimine el ruido y, que además, potencie los defectos que contenga.

Tomaremos como ejemplo la pieza cuyo mapa de vectores normales en RGB, se muestra en la Figura 3.6. Hemos escogido este ejemplo en concreto ya que presenta tres defectos en su superficie, una unión fría y dos inclusiones. Para una mejor visualización de los mismos hemos remarcado en color rojo la ubicación de cada uno de estos defectos. El número 1 señala una unión fría, mientras que el número 2 y el número 3 indican la posición de inclusiones.

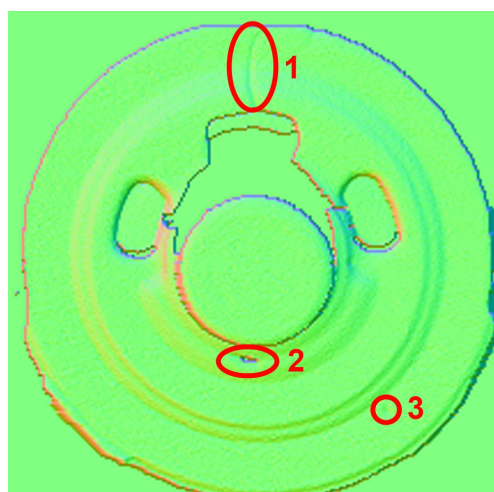


Figura 3.6: Mapa de normales en RGB de la pieza defectuosa.

1. Subproceso para cada uno de los modelos

- (a) Convertimos el mapa de normales codificado en RGB en una imagen en escala de grises. Con esto, conseguimos disminuir el gran volumen de ruido que presenta esta imagen en color.

2. Subproceso para cada una de las piezas defectuosas

- (a) Convertimos el mapa de normales codificado en RGB en una imagen en escala de grises. La razón principal por la que aplicamos

este paso, es disminuir el gran volumen de ruido que presenta esta imagen en color.

- (b) Aplicamos al resultado del paso anterior un filtro gaussiano de suavizado [GR02]. Con este filtro, minimizamos las variaciones no relevantes de la imagen.
- (c) Aplicamos un filtro de imagen, que calcula la diferencia píxel a píxel entre el resultado del paso anterior y cada uno de los modelos procesados. Así, obtenemos aquellas regiones de la imagen que presentan diferencias con respecto a cada uno de los modelos.
- (d) En este paso, agrupamos la información de todas las imágenes de diferencia que hemos obtenido en el paso anterior. Con este fin, aplicamos la intersección píxel a píxel de todas ellas, obteniendo una única imagen. La finalidad de esta intersección, es la de agrupar en una única imagen las diferencias comunes entre la imagen de la pieza defectuosa y cada uno de los modelos.
- (e) Una vez obtenida la intersección de las diferencias, la binarizamos, destacando, de esta forma, las regiones que son susceptibles de tener un defecto.
- (f) Finalmente, extraemos cada una de las regiones que han sido marcadas como consecuencia del proceso realizado. Además, eliminamos las regiones que no superen un determinado tamaño. En nuestro caso concreto, después de analizar los defectos existentes de forma empírica, hemos establecido un umbral de 3 píxeles de ancho por 3 píxeles de alto, que es el equivalente al tamaño mínimo para considerarlo como defecto en la inspección de calidad (0,5 mm).

Como resultado del proceso de segmentación que hemos descrito, obtenemos un conjunto de regiones entre las que encontramos partes de la superficie que presentan diferencias tolerables con respecto a los modelos, así como partes de la pieza con defectos que provocan el rechazo de la misma.

En el ejemplo con el que estamos trabajando, observamos que en la imagen que obtenemos como resultado del proceso de segmentación (véase la Figura 3.7), se han remarcado los tres defectos que presentaba la superficie de la pieza. Además de esto, han aparecido más regiones que indican diferencias comunes con respecto a los modelos. Aún así, en la inspección manual no han sido indicados como defectos.

Una vez aplicado este proceso a todas las piezas defectuosas, hemos obtenido 6.150 regiones en total. En la Tabla 3.2 podemos ver la cantidad de segmentos que pertenecen a cada una de las clases distintas: (i) tolerables, que son aquellos segmentos que no son motivo de rechazo; (ii) inclusiones, que delimitan una zona en la que hay una inclusión; (iii) uniones frías, que remarcan que en ese segmento existe una unión fría; y por último, (iv) las faltas de llenado, que indican la presencia de una falta de material en esa zona.

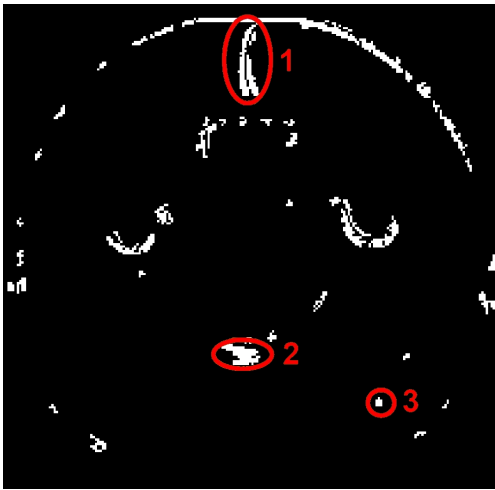


Figura 3.7: Imagen obtenida mediante el método de segmentación descrito. Re-markado en rojo los defectos.

Tabla 3.2: Número de regiones con cada tipo de defecto.

Tipo	Número de segmentos
<i>Tolerable</i>	5.686
<i>Inclusión</i>	392
<i>Unión fría</i>	17
<i>Falta de llenado</i>	55

Destacamos que el porcentaje de defectos detectados empleando este método de segmentación es del 59,70 %. La mayor parte de los defectos que no han sido localizados, se encuentran en zonas que coinciden con bordes de la pieza. Ésto, dificulta de forma considerable su detección, ya que es necesario un mayor número de capturas de la pieza mediante el sistema de visión artificial.

3.7 Sumario

En este capítulo hemos diseñado e implementado un sistema de visión artificial para capturar los datos de las superficies de las piezas. Este sistema está compuesto por: (i) un dispositivo de captura, que se basa en la triangulación láser; (ii) un brazo robótico, que automatiza el proceso de captura; y (iii) un equipo de procesamiento, que procesa los datos.

Posteriormente, con los datos capturados, hemos generado cuatro repre-

sentaciones distintas de la información: (i) la matriz de alturas, (ii) el mapa de alturas en escala de grises, (iii) la matriz de vectores normales, y por último, (iv) el mapa de vectores normales codificado en RGB.

Finalmente, hemos planteado un método de segmentación para delimitar las zonas de interés. Éstas, delimitan regiones en las que existe algún tipo de irregularidad en la superficie, pudiendo ser un defecto, o una zona tolerable. Con este método de segmentación y las representaciones generadas, hemos creado un conjunto de datos para realizar los distintos experimentos, basado en piezas reales provenientes de una fundición de hierro nodular.

«Es de sentido común elegir un método y probarlo. Si falla, admitirlo francamente y probar con otro. Pero, sobre todo, intentar algo.»

Franklin D. Roosevelt (1882–1945)

CAPÍTULO

4

Categorización de defectos superficiales

TAL y como hemos visto en la revisión del estado de la literatura en el Capítulo 2, existen diversos estudios que buscan detectar defectos superficiales. En nuestro caso, y como primer paso, en el Capítulo 3 hemos definido un método de segmentación con el que obtenemos aquellas regiones que pueden ser susceptibles de contener un defecto.

En este capítulo vamos a trabajar con el conjunto de datos formado por las regiones detectadas, extrayendo características de ellas y categorizándolas mediante algoritmos de aprendizaje automático supervisado.

El resto del capítulo continúa de la siguiente forma. La Sección 4.1 expone de un modo detallado los diferentes conjuntos de características que vamos a extraer de las regiones marcadas por el método de segmentación. La Sección 4.2 especifica los algoritmos de aprendizaje automático supervisado que hemos empleado para la experimentación. La sección 4.3 describe la validación empírica que hemos planteado para la evaluación del método propuesto. La sección 4.4 expone los resultados obtenidos a través de la experimentación realizada. Por último, la sección 4.5 resume las contribuciones realizadas en el este capítulo.

4.1 Conjuntos de características extraídas de los regiones detectadas

En esta sección vamos a estudiar los diferentes grupos de características que hemos extraído de las regiones. De esta forma, obtenemos las más significativas a la hora de categorizar mediante algoritmos de aprendizaje automático supervisado.

Atendiendo a su tipología podemos obtener los siguientes grupos de características: (i) las simples, que se corresponden con información existente en alguna de las representaciones de la región (sin realizar ningún tipo de procesamiento adicional para obtenerlas); (ii) las extraídas del BCLP (Best Crossing Line Profile), que busca minimizar la información irrelevante de la región; (iii) las obtenidas a partir de la transformada rápida de Fourier, que convierte la imagen de un dominio espacial o temporal, a un dominio frecuencial; (iv) y las extraídas de la matriz de co-ocurrencia, que componen método para representar la textura de la zona que estamos examinando.

4.1.1 Características simples

Denominamos características simples a aquéllas que podemos obtener a partir de las distintas representaciones que existen de las regiones. Así, no se requiere un procesamiento extra para generarlas. A continuación, agrupamos las distintas características pertenecientes a este grupo, que hemos extraído en base a su representación de origen:

- **Características extraídas de la imagen binarizada resultado del método de segmentación.** Esta es la imagen binarizada que hemos obtenido como resultado de aplicar el método de segmentación. De esta forma, tenemos marcado en color blanco la porción de la región marcada como potencialmente defectuosa y en negro el resto del segmento. Extraemos las siguientes características:
 - Alto, ancho, perímetro y área de la región.
 - Distancia euclidiana entre el centro de gravedad de la región y el origen de coordenadas.
 - Plenitud o *fullness* de la región, definida por la ecuación 4.1:

$$Fullness = \frac{\text{Área}}{(\text{Alto} * \text{Ancho})} \quad (4.1)$$

- **Características extraídas de la imagen binarizada convertida a imagen integral.** La imagen integral I de una imagen G se define como la imagen en la que la intensidad de un pixel en una posición determinada,

es igual a la suma de las intensidades de todos los píxeles que hay encima y a la izquierda de la posición dada en la imagen original G [VJ04]. Se extrae lo siguiente:

- Valor medio de los píxeles de la imagen integral.
- Sumatorio de los valores de los píxeles de la imagen integral.
- **Características extraídas de la matriz de alturas.** En este caso, extraemos dos regiones distintas. En la primera de ellas, obtenemos todos los píxeles de la matriz de alturas para la región extraída mediante el proceso de segmentación. En la segunda, nos quedamos únicamente con aquellos píxeles que hayan sido marcados en blanco en el segmento obtenido mediante el método de segmentación. En ambos casos, generamos un vector con los valores de ambas regiones y extraemos las siguientes características:
 - Sumatorio, media, varianza, desviación estándar, error estándar, mínimo, máximo, rango, mediana, entropía, sesgo y curtosis.
- **Características extraídas de las matrices de vectores normales.** Extraemos las regiones correspondientes a la delimitada por el método de segmentación. Como hemos especificado anteriormente, cada una de las matrices hace referencia a las componentes del vector normal en cada uno de los puntos (x,y,z) . Además, como en el caso anterior, extraemos por cada matriz dos regiones distintas, una obteniendo todos los valores y la otra con los valores marcados en blanco por el método de segmentación. A partir de cada una de ellas, generamos un vector con los valores correspondientes y extraemos las siguientes características:
 - Sumatorio, media, varianza, desviación estándar, error estándar, mínimo, máximo, rango, mediana, entropía, sesgo y curtosis.

4.1.2 Características del Best Crossing Line Profile

El concepto de *Crossing Line Profile* fue empleado por Mery [Mer03] para la detección de defectos en imágenes de piezas de aluminio fundido obtenidas mediante rayos X. A grandes rasgos, el algoritmo distingue entre las regiones que contienen defecto y las que contienen estructuras regulares sin tener una base de conocimiento previo.

En nuestro caso, nos hemos basado en este mismo concepto para extraer una serie de características, que posteriormente utilizaremos como entrada a un algoritmo de inteligencia artificial. El primer paso que damos es redimensionar la región detectada mediante el método de segmentación. Posteriormente, realizaremos este proceso para la región extraída del mapa de alturas

en escala de grises y para el mapa de vectores normales codificado en RGB. De esta forma normalizamos todas las regiones a un tamaño de 32 x 32 píxeles utilizando para ello un redimensionamiento basado en KNN.

En este sentido, definimos un *Crossing Line Profile* P_θ , como la función que identifica a la recta que atraviesa la región redimensionada r por su punto medio $(x/2, y/2)$, y que forma un ángulo θ con respecto al eje X . De la misma forma que Mery [Mer03], calculamos 8 perfiles P_θ , siendo $\theta = K\pi/8$ para los distintos valores de $K = 0, \dots, 7$.

Una vez calculados los 8 perfiles, elegimos aquel valor de k para el que los extremos del CLP sean más parecidos entre sí. A este perfil se le denomina *Best Crossing Line Profile* o BCLP y, con él podemos distinguir entre un defecto y un borde de la pieza examinada. Por último, extraemos ciertas características en base al perfil escogido en cada una de las representaciones. A continuación detallamos cada una de ellas.

- **Características obtenidas del BCLP de la región en el mapa de alturas en escala de grises.** Una vez calculado el BCLP para esta representación, extraemos los distintos valores de gris por los que pasa el perfil. Posteriormente obtenemos las siguientes características.
 - Sumatorio, media, varianza, desviación estándar, error estándar, mínimo, máximo, rango, mediana, entropía, sesgo y curtosis.
- **Características obtenidas del BCLP de la región en el mapa de vectores normales codificado en RGB.** En este caso, el BCLP se calcula para cada una de las componentes de la imagen RGB, obteniendo por cada una de ellas las siguientes características.
 - Sumatorio, media, varianza, desviación estándar, error estándar, mínimo, máximo, rango, mediana, entropía, sesgo y curtosis.

4.1.3 Características de la transformada rápida de Fourier

La transformada de Fourier es una transformación matemática que se usa en diversas áreas como las del procesamiento de señales, telecomunicaciones, o el procesamiento digital de imágenes entre otras [Bra99].

En esta misma línea, la transformada discreta de Fourier o DFT (por sus siglas en inglés Discrete Fourier Transform) es un tipo específico del análisis de Fourier que se encarga de convertir una función, que se encuentra en el dominio temporal o espacial, al dominio frecuencial.

En nuestro caso concreto, hemos empleado la transformada rápida de Fourier ó FFT (Fast Fourier Transform) que es una implementación eficiente de la transformada discreta de Fourier. Esta implementación se emplea en el procesamiento de imagen digital, y convierte la imagen de entrada (dominio espacial), en una imagen de salida (dominio frecuencial).

La transformada de Fourier descompone la imagen de entrada en sus componentes reales e imaginarias obteniendo de esta forma una representación de la misma en el dominio frecuencial, donde se cumple que el número de frecuencias en la representación del dominio frecuencial es igual al número de píxeles que tiene la imagen de entrada en el dominio espacial. A continuación mostramos la ecuación que se emplea para realizar la transformación (véase la Ecuación 4.2), así como la que se utiliza para realizar la transformación inversa, en la que la entrada es la imagen en el dominio frecuencial y el resultado la imagen en el dominio espacial (véase Ecuación 4.3).

$$F(p, q) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \cdot e^{-i2\pi(p \frac{m}{M} + q \frac{n}{N})} \quad (4.2)$$

$$f(m, n) = \frac{1}{M \cdot N} \cdot \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} F(p, q) \cdot e^{i2\pi(p \frac{m}{M} + q \frac{n}{N})} \quad (4.3)$$

donde $F(p, q)$ es el valor de la imagen en el dominio frecuencial correspondiente a las coordenadas (p, q) , $f(m, n)$ es el valor del píxel en la imagen del dominio espacial en las coordenadas (m, n) y M y N son las dimensiones de la imagen.

Esta transformación se la hemos aplicado a las distintas representaciones de la región potencialmente defectuosa. A continuación agrupamos las distintas características simples que hemos extraído en base a su representación de origen.

- **Características extraídas de la imagen transformada del mapa de alturas en escala de grises.** De la transformada obtenemos dos histogramas con los distintos niveles de gris. En el primero de ellos tenemos en cuenta todos los valores distintos de gris existentes, mientras que el segundo lo generamos sin tener en cuenta todos aquellos valores que sean iguales al color negro ($R = 0$, $G = 0$ y $B = 0$). Las características extraídas para cada uno de los histogramas de gris son las siguientes:
 - Máximo, media, mediana, mínimo, desviación estándar y entropía.
- **Características extraídas de la imagen transformada del mapa de vectores normales codificado en RGB.** En este caso concreto, hemos comenzado convirtiendo este mapa en escala de grises y posteriormente hemos obtenido la transformada correspondiente de Fourier. Finalmente, como en el caso anterior, obtenemos dos histogramas con los distintos niveles de gris. En el primero de ellos tenemos en cuenta todos los valores distintos de gris existentes, mientras que el segundo lo generamos sin tener en cuenta todos aquellos valores que sean iguales al color negro ($R = 0$, $G = 0$ y $B = 0$). Las características extraídas para cada uno de los histogramas de gris son las siguientes:

- Máximo, media, mediana, mínimo, desviación estándar y entropía.

4.1.4 Características de la matriz de co-ocurrencia

Las matrices de co-ocurrencia son un método para la medición de texturas en imágenes explicadas en la Sección 2.4.5.1. Una vez obtenida la matriz para el mapa de alturas en escala de grises, extraemos una serie de características definidas por Haralick et ál. [HSD73], entre las que encontramos: (i) el contraste, (ii) la disimilitud, (iii) la homogeneidad, (iv) el segundo momento angular, (v) la energía, (vi) la máxima probabilidad, (vii) la entropía, (viii) la media en X , (ix) la media en Y , (x) la varianza en X , (xi) la varianza en Y , (xii) la desviación estándar en X , (xiii) la desviación estándar en Y y (xiv) la correlación. A continuación especificamos las ecuaciones para calcular cada una de ellas:

$$\text{Contraste} = \sum_{i,j=0}^{N-1} P_{i,j}(i-j)^2 \quad (4.4)$$

$$\text{Disimilitud} = \sum_{i,j=0}^{N-1} P_{i,j}|i-j| \quad (4.5)$$

$$\text{Homogeneidad} = \sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+(i-j)^2} \quad (4.6)$$

$$\text{Segundo momento angular} = \sum_{i,j=0}^{N-1} (P_{i,j})^2 \quad (4.7)$$

$$\text{Energía} = \sqrt{\sum_{i,j=0}^{N-1} (P_{i,j})^2} \quad (4.8)$$

$$\text{Máxima probabilidad} = \text{argmax}(P_{i,j}) \quad (4.9)$$

$$\text{Entropía} = \sum_{i,j=0}^{N-1} P_{i,j}(-\ln P_{i,j}) \quad (4.10)$$

$$\text{Media en } X (\mu_i) = \sum_{i,j=0}^{N-1} i(P_{i,j}) \quad (4.11)$$

$$\text{Media en } Y (\mu_j) = \sum_{i,j=0}^{N-1} j(P_{i,j}) \quad (4.12)$$

$$\text{Varianza en } X (\sigma_i^2) = \sum_{i,j=0}^{N-1} P_{i,j}(i - \mu_i)^2 \quad (4.13)$$

$$\text{Varianza en } Y (\sigma_j^2) = \sum_{i,j=0}^{N-1} P_{i,j}(j - \mu_j)^2 \quad (4.14)$$

$$\text{Desviación típica en } X (\sigma_i) = \sqrt{\sigma_i^2} \quad (4.15)$$

$$\text{Desviación típica en } Y (\sigma_j) = \sqrt{\sigma_j^2} \quad (4.16)$$

$$\text{Correlación} = \sum_{i,j=0}^{N-1} P_{i,j} \left[\frac{(i - \mu_i)(j - \mu_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}} \right] \quad (4.17)$$

4.2 Categorización de defectos basada en aprendizaje automático supervisado

Los algoritmos de aprendizaje supervisado se caracterizan por emplear un conjunto de datos previamente etiquetados para obtener conclusiones y así poder predecir futuros comportamientos ante un contexto dado [KZP07]. En nuestro caso buscamos detectar y categorizar correctamente distintos defectos superficiales, basándonos para ello en un conjunto de defectos etiquetados que usamos como entrenamiento.

Los algoritmos que hemos empleado para la detección de defectos superficiales en las piezas de fundición son: (i) las redes bayesianas, basadas en el teorema de Bayes; (ii) las máquinas de soporte vectorial, que se basan en el uso de un hiperplano que divide el espacio en regiones; (iii) los k-vecinos más cercanos, que clasifican en base a la cercanía con respecto a datos ya etiquetados; y por último, (iv) los árboles de decisión, que se caracterizan por poder ser representados de una forma gráfica mediante un árbol.

4.2.1 Redes bayesianas

Las redes bayesianas se basan el teorema de Bayes [BP63] para realizar las predicciones de una nueva instancia desconocida. Este teorema se define como

un método estadístico que basa su razonamiento en un número de observaciones conocidas, para determinar la validez de una hipótesis dada. A su vez, ajusta las probabilidades tan pronto como dispone de nuevas observaciones.

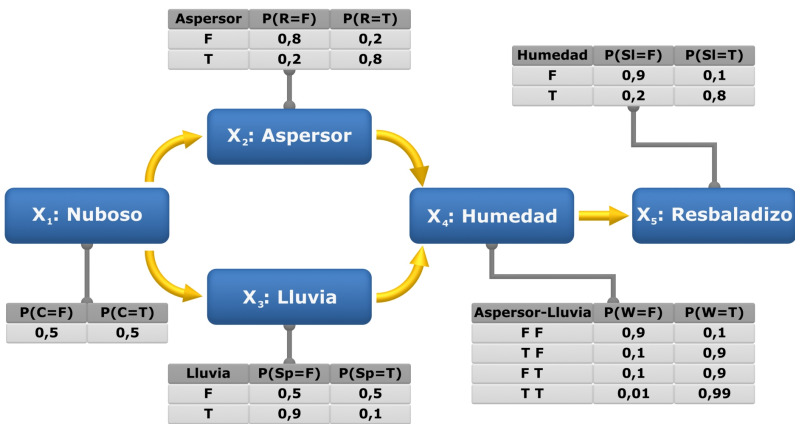


Figura 4.1: Ejemplo de clasificación mediante una red bayesiana.

Siguiendo su formulación clásica, podemos afirmar que dados dos eventos a los que denominaremos A y B , la probabilidad condicionada de que ocurra A habiéndose dado B , esto es, $P(A|B)$, se puede obtener mediante la Ecuación 4.18.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

(4.18)

Partiendo de este enfoque, se desarrollan las redes bayesianas (véase Figura 4.1), que son modelos probabilísticos para el análisis multivariable. Básicamente, son grafos acíclicos con una función de distribución probabilística asociada, donde los nodos representan variables y las aristas indican las dependencias condicionales que existen entre las distintas variables. Estos nodos pueden ser tanto una premisa como una conclusión [CGH97].

Los algoritmos de aprendizaje basados en redes bayesianas que hemos empleado son K2 [CH91], Tree Augmented Naïve o TAN [FGG97] y Naïve Bayes [Lew98].

4.2.2 Máquinas de soporte vectorial

Las máquinas de soporte vectorial también conocidas como SVM (Support Vector Machines) se basan en el uso de un hiperplano para dividir una representación n -dimensional de los datos en varias regiones y buscando de esta

forma la separación óptima entre los elementos existentes de cada clase. Este hiperplano busca maximizar el margen entre las regiones del espacio.

Formalmente, podemos definir que dado un conjunto de datos de entrenamiento $\mathcal{D}$, de tal forma que,

$$\mathcal{D} = \{(x_i, c_i) | x_i \in \mathbb{R}^p, c_i \in \{-1, 1\}\}_{i=1}^n \quad (4.19)$$

donde c_i indica la clase del punto x_i , pudiendo ser su valor -1 ó 1, y siendo a su vez, un vector de números reales en un espacio p -dimensional. El objetivo de este algoritmo es el de encontrar el hiperplano que maximice el margen m y que divida los puntos $c_i = 1$ de los $c_i = -1$ en el espacio formado por el conjunto de datos $\mathcal{D}$. La representación del hiperplano se realiza mediante los puntos x_i de tal forma que después de realizar el producto escalar entre x y w , y restado el valor de m , el valor resultante sea 0, siendo w un vector normal y ortogonal al hiperplano, o dicho de otro modo, los puntos x_i deben cumplir la ecuación 4.20

$$w \cdot x - m = 0 \quad (4.20)$$

De este modo tenemos dos hiperplanos, correspondiendo cada uno de ellos a cada clase de c_i , pudiéndose describir mediante la ecuación

$$c_i(w \cdot x_i - m) \geq 1, \forall x_i, 1 \leq i \leq n \quad (4.21)$$

En muchas ocasiones no se puede dividir el espacio mediante hiperplanos lineales, lo que hace necesario el uso de las denominadas *funciones del kernel*, con las que se pueden realizar clasificaciones polinomiales, radiales o sigmoideas.

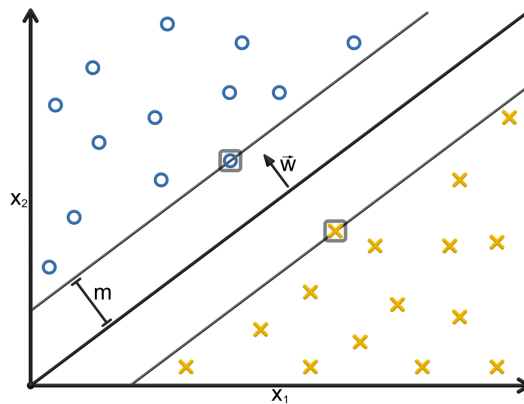


Figura 4.2: Ejemplo de clasificación mediante una máquina de soporte vectorial.

En esta tesis hemos empleado los siguientes tipos de *funciones del kernel*: (i) el polinomial [AW99], (ii) el polinomial normalizado [AW99], (iii) el RBF (Radial Basis Function) [AW99] y (iv) el Pearson VII [ÜMB06].

4.2.3 K vecinos más cercanos

El algoritmo K vecinos más cercanos o KNN (K Nearest Neighbours) es un método de clasificación que se basa en la cercanía con respecto a los elementos empleados en el espacio de entrenamiento para determinar la clase de un nuevo elemento desconocido [FHJ52].

La fase de entrenamiento comienza utilizando un conjunto de instancias $\mathcal{S} = \{s_1, s_2, \dots, s_{n-1}, s_n\}$, donde n representa el número de características de cada una de las instancias. Este conjunto de datos se representa en un espacio de entrenamiento como el que se muestra en la Figura 4.3.

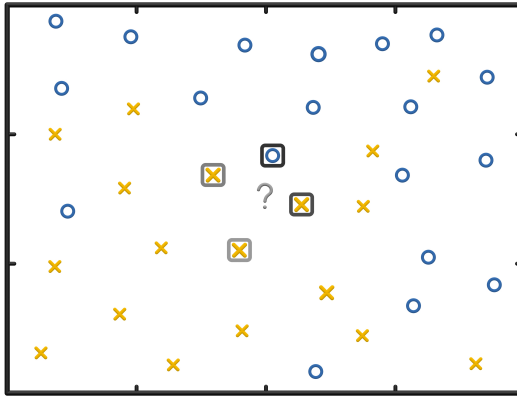


Figura 4.3: Ejemplo de clasificación mediante de K vecinos más cercanos.

Posteriormente, la fase de clasificación comienza con la llegada de una nueva instancia a clasificar. En este punto se calcula la distancia de esta instancia con respecto a las empleadas en la fase de entrenamiento. Esta distancia puede estar calculada en base a distintas métricas como por ejemplo, la distancia euclidiana, la distancia del coseno o la distancia de manhattan entre otras. En esta tesis hemos empleado la distancia euclidiana (véase la Ecuación 4.22) a la hora de aplicar el algoritmo de K vecinos más cercanos.

$$\sqrt{\sum_{i=0}^n (X_i - Y_i)^2} \quad (4.22)$$

Una vez calculadas las distancias en base a la métrica escogida, podemos emplear distintos métodos para determinar la clase de la instancia descono-

cida. La técnica más empleada es la de elegir la clase más común entre las K instancias más cercanas pertenecientes al espacio de entrenamiento y la instancia a clasificar.

4.2.4 Árboles de decisión

Una de las características más significativas de los árboles de decisión es que son un tipo dentro de los clasificadores de aprendizaje supervisado que pueden ser representados empleando una estructura en forma de árbol (véase la Figura 4.4). En este árbol los nodos interiores representan los estados posibles de las condiciones del problema. Por el contrario, los nodos finales constituyen las decisiones finales que puede devolver el clasificador.

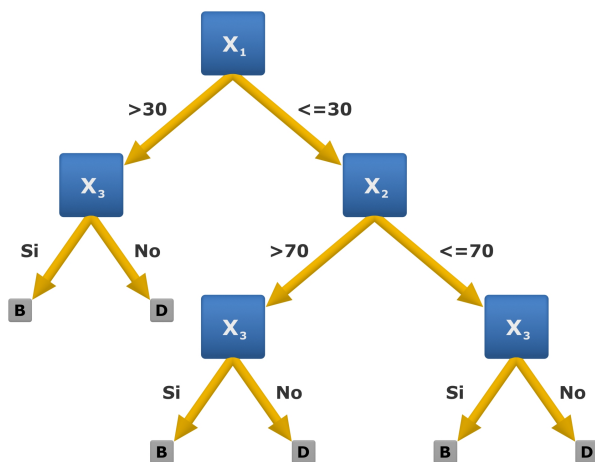


Figura 4.4: Ejemplo de clasificación mediante un árbol de decisión.

Formalmente, podemos definir un árbol de decisión como un grafo $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, donde $\mathcal{V}$ es un conjunto de nodos no vacío y $\mathcal{E}$ es un conjunto de aristas.

En este sentido, definimos un camino del árbol como la secuencia de aristas que recorre, pudiendo representarlo de la forma $(v_1, v_2), (v_2, v_3), \dots, (v_{n-1}, v_n)$. Por lo tanto, si tenemos en cuenta que (v, w) es una arista del árbol, podemos definir a v como el nodo padre de w , y a su vez a w como el nodo hijo de v .

Dentro del árbol existe un sólo nodo que no tiene nodo padre y al que denominamos nodo raíz, siendo un nodo interno cualquiera de los otros nodos del árbol y hojas todos los nodos finales del árbol.

Uno de los clasificadores basado en árboles de decisión que hemos utilizado en esta tesis es el algoritmo C4.5 [Qui93] (en la herramienta WEKA la

implementación de este algoritmo se denomina J48), que es una extensión del algoritmo ID3 [Qui86]. Su funcionamiento se basa en un concepto denominado *entropía de la información*, que calcula la incertidumbre asociada a una variable aleatoria. Este término en ocasiones va asociado a la *entropía de Shanon* [Sha51], que mide el valor esperado de la información dentro de un mensaje.

Otro método empleado para este trabajo es el de los bosques aleatorios [Bre01]. Este algoritmo genera varios árboles de decisión de forma aleatoria, combinándolos posteriormente como si fuesen un único clasificador. Concretamente, se genera un vector de entrada con cada uno de estos árboles, formado por un subconjunto de datos elegidos mediante una discriminación estocástica [Kle96].

Posteriormente, cada uno de los árboles generados realiza una clasificación. Para determinar el resultado final se emplea un sistema de votos en el que participan las clasificaciones de todos los árboles generados.

El principal inconveniente que presenta este clasificador es que es muy sensible al ruido que puedan presentar los nuevos datos [SLTW04], esto es, se adapta correctamente al conjunto de datos con los que realizamos el entrenamiento, pero pierde precisión en caso de que los nuevos datos que le van llegando contengan ruido.

4.3 Validación empírica de la categorización de defectos

En esta sección vamos analizar los resultados que hemos obtenido a través de los distintos experimentos que planteamos. Para ello, comenzamos con la especificación de la metodología de evaluación que seguimos en cada caso.

4.3.1 Metodología de evaluación

Para realizar los experimentos que planteamos, hemos utilizado la herramienta WEKA [G⁺95], ya que en ella se encuentran implementados y validados todos los algoritmos que hemos ido viendo a lo largo de la Sección 4.2.

En la Tabla 4.1, presentamos los algoritmos con los que hemos trabajado, junto con las distintas configuraciones de cada uno de ellos. Los clasificadores que no tienen una configuración indicada, han sido parametrizados con la configuración que traen por defecto en la herramienta propuesta.

Otro de los conceptos a destacar es que los resultados los hemos obtenido mediante la técnica de validación cruzada [B⁺06], que se basa en la división del conjunto de datos en K subconjuntos distintos, empleándose una parte de cada uno de estos subconjuntos para entrenar el modelo, y la otra para

Tabla 4.1: Algoritmos de aprendizaje supervisado empleados, junto con sus configuraciones.

Algoritmo	Configuración empleada
Redes bayesianas	K2 y TAN
Naïve Bayes	
SVM	Polinomial, Polinomial normalizado, Pearson VII y RBF
KNN	$K = 1, K = 2, K = 3, K = 4$ y $K = 5$
J48	
Random Forest	$N = 25, N = 50, N = 75$ y $N = 100$

probarlo. En nuestro caso concreto, le hemos asignado a K el valor 10, con lo que hemos dividido el conjunto de datos en 10 subconjuntos distintos. En cada uno de estos subconjuntos, se ha utilizado el 90 % de los datos para entrenar el modelo y el 10 % restante para probarlo.

Como último apunte y debido a que el conjunto de datos del que disponemos no se encuentra balanceado (véase la Tabla 3.2), hemos empleado una técnica denominada SMOTE (por sus siglas en inglés Synthetic Minority Over-sampling TEchnique)[CBHK02]. Ésta, genera muestras sintéticas de las clases minoritarias, balanceando de esta forma el conjunto de datos de entrenamiento.

4.3.2 Medición de los resultados

En este punto vamos a ver las diferentes métricas que hemos empleado para evaluar cada uno de los modelos generados en los experimentos. Estas métricas dependen del tipo de experimento que realicemos. En primer lugar para los experimentos de categorización de defectos empleamos: (i) la tasa agregada de verdaderos positivos, (ii) la tasa agregada de falsos positivos (iii) la precisión y (iv) el área bajo la curva ROC. En el caso de los experimentos realizados para la detección de defectos usamos: (i) la tasa de verdaderos positivos, (ii) la tasa de falsos positivos, (iii) la precisión y (iv) el área bajo la curva ROC.

4.3.2.1 Tasa de verdaderos positivos

La tasa de verdaderos positivos, también conocidos como TPR (True Positive Rate) o sensibilidad, se calcula dividiendo el número de regiones defectuosas clasificadas como defectuosas TP , entre el número total de regiones defectuosas ($TP + FN$).

$$TPR = \frac{TP}{TP + FN}$$

(4.23)

4.3.2.2 Tasa de falsos positivos

La tasa de falsos positivos, también conocida como *FPR* (False Positive Rate) o especificidad, se calcula dividiendo el número de regiones defectuosas clasificadas como tolerables *TN*, entre el número total de regiones tolerables $FP + TN$.

$$FPR = \frac{FP}{FP + TN} \quad (4.24)$$

4.3.2.3 Tasa agregada de verdaderos positivos

La tasa agregada de verdaderos positivos, que denominaremos *ATPR*, se calcula restando a 1 la tasa de falsos negativos. Con esta medida conseguimos obtener, en el caso de la categorización de defectos, la tasa de regiones defectuosas (sin tener en cuenta el tipo de defecto) que han sido clasificadas como defectuosas.

$$ATPR = 1 - \frac{FN}{FN + TP} \quad (4.25)$$

4.3.2.4 Tasa agregada de falsos positivos

La tasa agregada de falsos positivos, que denominaremos *AFPR*, se calcula restando a 1 la tasa de verdaderos negativos. De esta forma, obtenemos la tasa de regiones tolerables que han sido categorizadas como cualquiera de los tipos de error definidos.

$$AFPR = 1 - \frac{TN}{TN + FP} \quad (4.26)$$

4.3.2.5 Precisión

Para calcular la precisión, dividimos los aciertos totales (suma de los verdaderos positivos y verdaderos negativos), entre todas las regiones existentes.

$$\text{Precisión}(\%) = \frac{TP + TN}{TN + FP + TN + FN} \quad (4.27)$$

4.3.2.6 Área bajo la curva ROC

El área bajo la curva ROC o *AUC* (Area Under ROC Curve) establece la relación entre los falsos positivos *FP* (regiones tolerables que han sido clasificadas como defectuosas) y los falsos negativos *FN* (regiones defectuosas clasificadas como tolerables) [SKM09].

4.3.3 Resultados obtenidos mediante la experimentación

En esta sección, exponemos los resultados que hemos obtenido a través de la experimentación realizada. Para ello, planteamos dos estrategias distintas basadas en diferentes enfoques para categorizar las regiones. Estas dos estrategias son: (i) la categorización en dos fases, que se basa en realizar una primera clasificación basada en regiones tolerables y regiones defectuosas y, posteriormente categorizar las regiones defectuosas inclusiones, uniones frías o faltas de llenado; y la (ii) la categorización directa, que se basa en determinar si la región es tolerable o si tiene inclusión, unión fría o falta de llenado de forma directa.

Con estos dos enfoques buscamos la mejor estrategia para la categorización de regiones. A continuación, mostramos los distintos resultados que hemos obtenido en cada uno de los casos, junto con la comparativa de los diferentes grupos de características empleadas, así como los tipos de clasificadores de aprendizaje automático supervisado empleados.

4.3.3.1 Categorización en dos fases

Esta estrategia se compone de dos fases. En la primera de ellas partimos de un conjunto de datos en el que están presentes todas las regiones tolerables y todas las defectuosas. Estas últimas se encuentran agrupadas, no distinguiéndose el tipo de defecto que contienen. En la segunda fase, el conjunto de datos, estará formado únicamente por las regiones defectuosas, y se categorizarán en base al tipo de defecto que contienen.

- Fase 1 - Características del *Best Crossing Line Profile*.

Tabla 4.2: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando las características extraídas del *Best Crossing Line Profile* para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	<i>TPR</i>	<i>FPR</i>	<i>AUC</i>
<i>RB: K2</i>	79,00	0,7764	0,2100	0,8656
<i>RB: TAN</i>	92,55	0,4177	0,0745	0,8857
<i>Naïve Bayes</i>	70,08	0,8609	0,2992	0,8348
<i>SVM: Polinomial</i>	75,87	0,8703	0,2413	0,8753
<i>SVM: Polinomial norm.</i>	77,45	0,8668	0,2255	0,8894
<i>SVM: Pearson VII</i>	90,28	0,6948	0,0972	0,8995
<i>SVM: RBF</i>	73,75	0,8661	0,2625	0,8695
<i>KNN K=1</i>	90,79	0,6284	0,0921	0,7618
<i>KNN K=2</i>	92,22	0,5703	0,0778	0,8248
<i>KNN K=3</i>	90,03	0,7027	0,0997	0,8526
<i>KNN K=4</i>	90,94	0,6672	0,0906	0,8630
<i>KNN K=5</i>	89,27	0,7392	0,1073	0,8718
<i>AD: J48</i>	89,89	0,6373	0,1011	0,7631
<i>AD: Random Forest N=10</i>	93,17	0,6404	0,0683	0,9015
<i>AD: Random Forest N=25</i>	92,97	0,6789	0,0703	0,9158
<i>AD: Random Forest N=50</i>	93,23	0,6712	0,0677	0,9201
<i>AD: Random Forest N=75</i>	93,17	0,6828	0,0683	0,9221
<i>AD: Random Forest N=100</i>	93,27	0,6790	0,0673	0,9229

Para este conjunto de características (véase la Tabla 4.2), los clasificadores se han comportado, en general, con precisiones superiores al 90% y áreas bajo la curva ROC mayores de 0,85, en su mayoría. En concreto, *Random Forest* es el clasificador que obtiene el resultado más destacado con todas las configuraciones probadas. El clasificador SVM es el que peores resultados ha obtenido para este grupo de características junto con los clasificadores bayesianos, exceptuando las redes bayesianas con TAN, que obtienen unos resultados similares a *Random Forest*.

• Fase 1 - Características de la transformada rápida de Fourier.

Tabla 4.3: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando las características de la transformada rápida de Fourier para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	TPR	FPR	AUC
<i>RB: K2</i>	77,66	0,8201	0,2234	0,8700
<i>RB: TAN</i>	86,71	0,7443	0,1329	0,8822
<i>Naïve Bayes</i>	67,58	0,8552	0,3242	0,7973
<i>SVM: Polinomial</i>	74,40	0,9031	0,2560	0,8608
<i>SVM: Polinomial norm.</i>	75,19	0,8944	0,2481	0,8674
<i>SVM: Pearson VII</i>	82,38	0,8645	0,1762	0,8914
<i>SVM: RBF</i>	73,97	0,8555	0,2603	0,8432
<i>KNN K=1</i>	88,07	0,6149	0,1193	0,7492
<i>KNN K=2</i>	90,18	0,5353	0,0982	0,8179
<i>KNN K=3</i>	87,58	0,6965	0,1242	0,8479
<i>KNN K=4</i>	88,80	0,6545	0,1120	0,8623
<i>KNN K=5</i>	86,81	0,7539	0,1319	0,8699
<i>AD: J48</i>	88,51	0,6268	0,1149	0,7593
<i>AD: Random Forest N=10</i>	91,10	0,5907	0,0890	0,8775
<i>AD: Random Forest N=25</i>	90,80	0,6370	0,0920	0,8921
<i>AD: Random Forest N=50</i>	91,10	0,6365	0,0890	0,8966
<i>AD: Random Forest N=75</i>	90,98	0,6437	0,0902	0,8983
<i>AD: Random Forest N=100</i>	91,05	0,6376	0,0895	0,8992

Con el grupo de características extraídas de la transformada rápida de Fourier (véase la Tabla 4.3), los clasificadores han obtenido, por norma general, unos valores de más del 85 % de precisión y áreas bajo la curva ROC mayores de 0,85. Particularmente, el clasificador *Random Forest* configurado con un número de árboles igual a 100 ha destacado sobre el resto de clasificadores con una precisión del 91,05 %, un TPR de 0,6376, un FPR de tan sólo un 0,0895 y un área bajo la curva ROC de 0,8992. Por contra, el clasificador SVM ha sido el que peores resultados ha obtenido para este conjunto de características, distando mucho del resto de clasificadores.

- Fase 1 - Características de la matriz de co-ocurrencia.

Tabla 4.4: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando las características de la matriz de co-ocurrencia para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	TPR	FPR	AUC
<i>RB: K2</i>	63,93	0,6967	0,3607	0,7291
<i>RB: TAN</i>	80,08	0,5212	0,1992	0,7305
<i>Naïve Bayes</i>	82,66	0,3441	0,1734	0,7162
<i>SVM: Polinomial</i>	63,33	0,7302	0,3667	0,7665
<i>SVM: Polinomial norm.</i>	69,45	0,6966	0,3055	0,7778
<i>SVM: Pearson VII</i>	75,22	0,6679	0,2478	0,7877
<i>SVM: RBF</i>	62,63	0,7314	0,3737	0,7396
<i>KNN K=1</i>	82,04	0,4975	0,1796	0,6824
<i>KNN K=2</i>	86,68	0,4110	0,1332	0,7213
<i>KNN K=3</i>	80,64	0,5541	0,1936	0,7391
<i>KNN K=4</i>	83,78	0,4995	0,1622	0,7519
<i>KNN K=5</i>	79,39	0,5802	0,2061	0,7603
<i>AD: J48</i>	81,91	0,5023	0,1809	0,6800
<i>AD: Random Forest N=10</i>	87,68	0,4501	0,1232	0,7582
<i>AD: Random Forest N=25</i>	87,12	0,4867	0,1288	0,7725
<i>AD: Random Forest N=50</i>	87,73	0,4797	0,1227	0,7797
<i>AD: Random Forest N=75</i>	87,56	0,4863	0,1244	0,7813
<i>AD: Random Forest N=100</i>	87,74	0,4815	0,1226	0,7820

La Tabla 4.4 muestra los resultados obtenidos para el conjunto de características obtenidas a partir de la matriz de co-ocurrencia, donde los distintos clasificadores arrojan unas precisiones cercanas al 90% y unas áreas bajo la curva ROC inferiores a 0,80. Profundizando, el resultado más destacable es el de *Random Forest* configurado con 100 árboles. Por contra, los clasificadores que peores resultados han obtenido para el grupo de características dado han sido los bayesianos y los SVMs.

- Fase 1 - Características simples.

Tabla 4.5: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando las características simples para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	TPR	FPR	AUC
<i>RB: K2</i>	93,14	0,7152	0,0686	0,8908
<i>RB: TAN</i>	95,81	0,5339	0,0419	0,9364
<i>Naïve Bayes</i>	85,96	0,8324	0,1404	0,9002
<i>SVM: Polinomial</i>	93,45	0,8382	0,0655	0,9609
<i>SVM: Polinomial norm.</i>	94,39	0,7965	0,0561	0,9621
<i>SVM: Pearson VII</i>	97,27	0,8164	0,0273	0,9742
<i>SVM: RBF</i>	92,79	0,8393	0,0721	0,9588
<i>KNN K=1</i>	92,78	0,2672	0,0722	0,5975
<i>KNN K=2</i>	93,00	0,1692	0,0700	0,6366
<i>KNN K=3</i>	93,24	0,2143	0,0676	0,6768
<i>KNN K=4</i>	93,01	0,1585	0,0699	0,7123
<i>KNN K=5</i>	93,11	0,1853	0,0689	0,7415
<i>AD: J48</i>	95,12	0,7403	0,0488	0,8487
<i>AD: Random Forest N=10</i>	96,76	0,7582	0,0324	0,9563
<i>AD: Random Forest N=25</i>	96,93	0,7920	0,0307	0,9687
<i>AD: Random Forest N=50</i>	97,04	0,7909	0,0296	0,9721
<i>AD: Random Forest N=75</i>	97,05	0,7958	0,0295	0,9735
<i>AD: Random Forest N=100</i>	97,06	0,7932	0,0294	0,9744

Para este conjunto de características (véase la Tabla 4.5), que hemos denominado simples, los clasificadores se han comportado, de forma general, con precisiones superiores al 90% y áreas bajo la curva ROC mayores de 0,95. A diferencia de los anteriores conjuntos de características, para este grupo, todos los clasificadores utilizados han mejorado de forma significativa sus resultados. En concreto, destaca el resultado obtenido por Random Forest configurado con un número de árboles igual a 100, con una precisión del 97,06%, un TPR de 0,7932, un FPR de 0,0294 y un área bajo la curva ROC de 0,9744. En la misma línea, el clasificador SVM con un kernel Pearson VII muestra unos resultados similares a los de *Random Forest* con una precisión de 97,27%, un TPR de 0,8164, un FPR 0,0273, y un área bajo la curva ROC de 0,9742.

Teniendo en cuenta que los mejores resultados obtenidos hasta ahora se han conseguido utilizando el conjunto de características simples, vamos a combinar éste con los distintos grupos empleados hasta el momento.

- **Fase 1 - Características simples combinadas con las características del *Best Crossing Line Profile*.**

Tabla 4.6: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando las características del *Best Crossing Line Profile* combinadas con las características simples para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	TPR	FPR	AUC
<i>RB: K2</i>	92,99	0,6779	0,0701	0,8695
<i>RB: TAN</i>	94,34	0,3192	0,0566	0,8996
<i>Naïve Bayes</i>	84,92	0,8304	0,1508	0,8900
<i>SVM: Polinomial</i>	94,02	0,8201	0,0598	0,9599
<i>SVM: Polinomial norm.</i>	94,35	0,7844	0,0565	0,9592
<i>SVM: Pearson VII</i>	96,85	0,7793	0,0315	0,9726
<i>SVM: RBF</i>	93,38	0,8080	0,0662	0,9578
<i>KNN K=1</i>	92,76	0,2745	0,0724	0,6011
<i>KNN K=2</i>	93,01	0,1714	0,0699	0,6426
<i>KNN K=3</i>	93,24	0,2341	0,0676	0,6769
<i>KNN K=4</i>	93,15	0,1867	0,0685	0,7079
<i>KNN K=5</i>	93,26	0,2237	0,0674	0,7347
<i>AD: J48</i>	95,08	0,7285	0,0492	0,8418
<i>AD: Random Forest N=10</i>	96,72	0,7433	0,0328	0,9574
<i>AD: Random Forest N=25</i>	96,92	0,7807	0,0308	0,9684
<i>AD: Random Forest N=50</i>	97,02	0,7806	0,0298	0,9726
<i>AD: Random Forest N=75</i>	97,06	0,7842	0,0294	0,9740
<i>AD: Random Forest N=100</i>	97,09	0,7858	0,0291	0,9746

Con el grupo de características compuestas por la combinación de las características simples y las obtenidas del *Best Crossing Line Profile* (véase la Tabla 4.6), los clasificadores han mantenido, por lo general, la misma tendencia que con las características simples. En concreto, los valores para la precisión superan el 90 % y para el área bajo la curva ROC son de más de 0,95. A modo particular destaca el clasificador *Random Forest* configurado con 100 árboles, junto con SVM con un kernel Pearson VII. Por otro lado, los clasificadores bayesianos muestran un buen comportamiento, a excepción de Naïve Bayes, que se ha convertido en el clasificador con los resultados más bajos.

- **Fase 1 - Características simples combinadas con las características de la transformada rápida de Fourier.**

Tabla 4.7: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando las características de la transformada rápida de Fourier combinadas con las características simples para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	TPR	FPR	AUC
<i>RB: K2</i>	91,32	0,7373	0,0868	0,8836
<i>RB: TAN</i>	95,36	0,4741	0,0464	0,9271
<i>Naïve Bayes</i>	84,56	0,8471	0,1544	0,8983
<i>SVM: Polinomial</i>	93,62	0,8308	0,0638	0,9604
<i>SVM: Polinomial norm.</i>	94,48	0,7940	0,0552	0,9612
<i>SVM: Pearson VII</i>	97,31	0,8146	0,0269	0,9770
<i>SVM: RBF</i>	93,01	0,8290	0,0699	0,9576
<i>KNN K=1</i>	92,93	0,2604	0,0707	0,5949
<i>KNN K=2</i>	93,00	0,1700	0,0700	0,6323
<i>KNN K=3</i>	93,15	0,2135	0,0685	0,6707
<i>KNN K=4</i>	93,02	0,1617	0,0698	0,7105
<i>KNN K=5</i>	93,10	0,1869	0,0690	0,7383
<i>AD: J48</i>	95,14	0,7375	0,0486	0,8425
<i>AD: Random Forest N=10</i>	96,71	0,7544	0,0329	0,9593
<i>AD: Random Forest N=25</i>	96,87	0,7866	0,0313	0,9684
<i>AD: Random Forest N=50</i>	97,11	0,7921	0,0289	0,9726
<i>AD: Random Forest N=75</i>	97,08	0,7945	0,0292	0,9740
<i>AD: Random Forest N=100</i>	97,10	0,7939	0,0290	0,9741

La Tabla 4.7 muestra los resultados obtenidos para el conjunto de características obtenidas a partir de la combinación de las características simples y de las obtenidas a partir de la transformada rápida de Fourier, donde los distintos clasificadores arrojan unas precisiones cercanas al 95 % y unas áreas bajo la curva ROC superiores a 0,90. Profundizando, el resultado más destacable es el de SVM con un kernel Pearson VII, así como *Random Forest* configurado con 100 árboles. Por contra, los clasificadores que peores resultados han obtenido para el grupo de características dado han sido los bayesianos a excepción de la red bayesiana entrenada con TAN.

- **Fase 1 - Características simples combinadas con las características de la matriz de co-ocurrencia.**

Tabla 4.8: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando las características de la matriz de co-ocurrencia combinadas con las características simples para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	TPR	FPR	AUC
<i>RB: K2</i>	92,59	0,6918	0,0741	0,8801
<i>RB: TAN</i>	95,50	0,4800	0,0450	0,9274
<i>Naïve Bayes</i>	86,35	0,8317	0,1365	0,8998
<i>SVM: Polinomial</i>	93,76	0,8262	0,0624	0,9619
<i>SVM: Polinomial norm.</i>	94,45	0,7892	0,0555	0,9620
<i>SVM: Pearson VII</i>	97,29	0,8196	0,0271	0,9770
<i>SVM: RBF</i>	92,98	0,8323	0,0702	0,9591
<i>KNN K=1</i>	92,80	0,2731	0,0720	0,6006
<i>KNN K=2</i>	92,99	0,1686	0,0701	0,6372
<i>KNN K=3</i>	93,20	0,2167	0,0680	0,6809
<i>KNN K=4</i>	93,01	0,1613	0,0699	0,7146
<i>KNN K=5</i>	93,15	0,1926	0,0685	0,7453
<i>AD: J48</i>	95,11	0,7334	0,0489	0,8367
<i>AD: Random Forest N=10</i>	96,68	0,7508	0,0332	0,9565
<i>AD: Random Forest N=25</i>	96,86	0,7907	0,0314	0,9682
<i>AD: Random Forest N=50</i>	96,99	0,7896	0,0301	0,9728
<i>AD: Random Forest N=75</i>	96,99	0,7928	0,0301	0,9738
<i>AD: Random Forest N=100</i>	97,02	0,7922	0,0298	0,9749

Para este conjunto de características (véase la Tabla 4.2), formada por la combinación de características simples y las extraídas de la matriz de co-ocurrencia, los clasificadores se han comportado, en general, con precisiones superiores al 92 % y áreas bajo la curva ROC mayores de 0,90, en su mayoría. Concretamente, SVM con kernel Pearson VII es el clasificador que obtiene el resultado más destacado, junto con *Random Forest* configurado con más de 25 árboles. El clasificador Naïve Bayes es el que peores resultados ha obtenido para este grupo de características junto con KNN.

- **Fase 1 - Todas las características.**

Tabla 4.9: Resultados obtenidos en términos de precisión (en porcentaje), tasa de verdaderos positivos (*TPR*), tasa de falsos positivos (*FPR*) y área bajo la curva ROC (*AUC*) empleando todas las características obtenidas para la Fase 1 (tolerable / defectuosa).

Algoritmo	Prec.(%)	TPR	FPR	AUC
<i>RB: K2</i>	91,44	0,6881	0,0856	0,8584
<i>RB: TAN</i>	93,95	0,2688	0,0605	0,8637
<i>Naïve Bayes</i>	84,11	0,8415	0,1589	0,8909
<i>SVM: Polinomial</i>	94,24	0,8067	0,0576	0,9601
<i>SVM: Polinomial norm.</i>	94,50	0,7845	0,0550	0,9590
<i>SVM: Pearson VII</i>	96,88	0,7789	0,0312	0,9752
<i>SVM: RBF</i>	93,70	0,8021	0,0630	0,9567
<i>KNN K=1</i>	92,79	0,2649	0,0721	0,5964
<i>KNN K=2</i>	93,05	0,1762	0,0695	0,6439
<i>KNN K=3</i>	93,20	0,2369	0,0680	0,6793
<i>KNN K=4</i>	93,16	0,1848	0,0684	0,7091
<i>KNN K=5</i>	93,29	0,2239	0,0671	0,7379
<i>AD: J48</i>	94,87	0,7207	0,0513	0,8322
<i>AD: Random Forest N=10</i>	96,71	0,7416	0,0329	0,9557
<i>AD: Random Forest N=25</i>	96,84	0,7781	0,0316	0,9682
<i>AD: Random Forest N=50</i>	96,96	0,7767	0,0304	0,9725
<i>AD: Random Forest N=75</i>	96,97	0,7819	0,0303	0,9735
<i>AD: Random Forest N=100</i>	97,03	0,7806	0,0297	0,9740

Con el grupo de características formado por la combinación de todas las empleadas hasta el momento (véase la Tabla 4.9), los clasificadores han obtenido, por norma general, unos valores de más del 92 % de precisión y áreas bajo la curva ROC mayores de 0,90. Particularmente, el clasificador *Random Forest* configurado con un número de árboles igual a 100, ha destacado sobre el resto de clasificadores con una precisión del 91,05 %, un TPR de 0,6376, un FPR de tan sólo un 0,0895 y un área bajo la curva ROC de 0,8992. Otro de los clasificadores destacados es SVM con kernel Pearson VII, que presenta una precisión del 96,88 %, un TPR de 0,7789, un FPR de 0,0312 y un área bajo la curva ROC de 0,9752. El clasificador Naïve Bayes ha arrojado los peores resultados para esta configuración.

• Fase 2 - Características extraídas del *Best Crossing Line Profile*.

Tabla 4.10: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (AUC) empleando las características del *Best Crossing Line Profile* para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
RB: K2	82,74	0,8423
RB: TAN	86,21	0,8441
Naïve Bayes	83,90	0,7597
SVM: Polinomial	77,04	0,8019
SVM: Polinomial norm.	78,81	0,8316
SVM: Pearson VII	83,06	0,8642
SVM: RBF	76,21	0,8016
KNN K=1	83,06	0,7151
KNN K=2	85,47	0,7378
KNN K=3	81,73	0,7610
KNN K=4	83,36	0,7691
KNN K=5	80,67	0,7666
AD: J48	78,81	0,7317
AD: Random Forest N=10	83,55	0,8617
AD: Random Forest N=25	82,81	0,8751
AD: Random Forest N=50	82,88	0,8780
AD: Random Forest N=75	82,75	0,8777
AD: Random Forest N=100	83,12	0,8782

La Tabla 4.10 muestra los resultados obtenidos para el conjunto de características obtenidas a partir del *Best Crossing Line Profile*, donde los distintos clasificadores arrojan unas precisiones en torno al 80% y unas áreas bajo la curva ROC cercanas a 0,85. Profundizando, los resultados más destacables son los obtenidos por SVM con kernel Pearson VII y, por *Random Forest* configurado con 100 árboles. Por su parte, entre los clasificadores bayesianos, la red bayesiana entrenada con TAN también arroja un buen resultado. Por contra, el clasificador que peores resultados ha obtenido para el grupo de características dado ha sido KNN con todos los valores probados para la K.

- **Fase 2 - Características extraídas de la transformada rápida de Fourier.**

Tabla 4.11: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (*AUC*) empleando las características de la transformada rápida de Fourier para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
<i>RB: K2</i>	82,87	0,7307
<i>RB: TAN</i>	80,58	0,7512
<i>Naïve Bayes</i>	85,70	0,7244
<i>SVM: Polinomial</i>	59,68	0,7518
<i>SVM: Polinomial norm.</i>	60,01	0,7757
<i>SVM: Pearson VII</i>	65,73	0,7734
<i>SVM: RBF</i>	69,22	0,7349
<i>KNN K=1</i>	78,25	0,6779
<i>KNN K=2</i>	82,86	0,7228
<i>KNN K=3</i>	78,89	0,7465
<i>KNN K=4</i>	80,54	0,7516
<i>KNN K=5</i>	78,16	0,7622
<i>AD: J48</i>	77,02	0,6897
<i>AD: Random Forest N=10</i>	81,18	0,7754
<i>AD: Random Forest N=25</i>	81,51	0,7845
<i>AD: Random Forest N=50</i>	81,85	0,7905
<i>AD: Random Forest N=75</i>	81,55	0,7924
<i>AD: Random Forest N=100</i>	81,55	0,7913

Para este conjunto de características (véase la Tabla 4.11), extraídas de la transformada rápida de Fourier, los clasificadores se han comportado, en general, con precisiones superiores al 80 % y áreas bajo la curva ROC mayores de 0,75. En concreto, *Random Forest* es el clasificador que obtiene el resultado más destacado configurado con más de 25 árboles. Por otro lado, el clasificador SVM es el que peores resultados ha obtenido para este grupo de características junto con el clasificador KNN con los distintos valores de *K* probados. Por su parte, las redes bayesianas han obtenido unos resultados cercanos los de *Random Forest*.

- Fase 2 - Características extraídas de la matriz de co-ocurrencia.

Tabla 4.12: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (*AUC*) empleando las características de la matriz de co-ocurrencia para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
<i>RB: K2</i>	68,82	0,7641
<i>RB: TAN</i>	71,19	0,7194
<i>Naïve Bayes</i>	73,79	0,7574
<i>SVM: Polinomial</i>	64,52	0,7560
<i>SVM: Polinomial norm.</i>	60,93	0,7643
<i>SVM: Pearson VII</i>	69,63	0,7643
<i>SVM: RBF</i>	65,73	0,7483
<i>KNN K=1</i>	73,34	0,6637
<i>KNN K=2</i>	79,09	0,7034
<i>KNN K=3</i>	72,76	0,7280
<i>KNN K=4</i>	76,23	0,7379
<i>KNN K=5</i>	71,55	0,7429
<i>AD: J48</i>	68,60	0,6674
<i>AD: Random Forest N=10</i>	77,33	0,7325
<i>AD: Random Forest N=25</i>	77,61	0,7376
<i>AD: Random Forest N=50</i>	77,75	0,7390
<i>AD: Random Forest N=75</i>	77,97	0,7394
<i>AD: Random Forest N=100</i>	77,95	0,7407

Con el grupo de características extraídas de la matriz de co-ocurrencia (véase la Tabla 4.12), los clasificadores han obtenido, por norma general, unos valores de menos del 80 % de precisión y áreas bajo la curva ROC menores de 0,75. Particularmente, el clasificador *Random Forest*, con todas las configuraciones, ha destacado sobre el resto con una precisión superior al 77%, y un área bajo la curva ROC de más 0,73. Por contra, el clasificador SVM ha sido el que peores resultados ha obtenido para este conjunto de características, distando mucho del resto de clasificadores. Asimismo, Naïve Bayes arroja los mejores resultados entre los clasificadores bayesianos con una precisión de 73,79% y un área bajo la curva ROC de 0,7574.

- Fase 2 - Características simples.

Tabla 4.13: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (AUC) empleando las características simples para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
<i>RB: K2</i>	84,18	0,8290
<i>RB: TAN</i>	89,25	0,8986
<i>Naïve Bayes</i>	86,10	0,7876
<i>SVM: Polinomial</i>	90,55	0,9373
<i>SVM: Polinomial norm.</i>	86,40	0,9289
<i>SVM: Pearson VII</i>	88,22	0,9097
<i>SVM: RBF</i>	83,08	0,9044
<i>KNN K=1</i>	89,77	0,7587
<i>KNN K=2</i>	89,01	0,8262
<i>KNN K=3</i>	88,45	0,8639
<i>KNN K=4</i>	88,04	0,8682
<i>KNN K=5</i>	88,28	0,8778
<i>AD: J48</i>	88,70	0,8492
<i>AD: Random Forest N=10</i>	90,24	0,9515
<i>AD: Random Forest N=25</i>	90,91	0,9584
<i>AD: Random Forest N=50</i>	91,25	0,9612
<i>AD: Random Forest N=75</i>	91,36	0,9629
<i>AD: Random Forest N=100</i>	91,30	0,9638

La Tabla 4.13 muestra los resultados obtenidos para el conjunto de características simples, donde como ocurría en la primera fase, los distintos clasificadores arrojan unas precisiones cercanas al 90% y unas áreas bajo la curva ROC superiores a 0,90. Específicamente, el resultado más destacable es el de *Random Forest* y, en concreto, configurado con más de 50 árboles. Por su parte, destaca también SVM configurado con kernel polinomial, que ha obtenido una precisión del 90,55% y un área bajo la curva ROC de 0,9373. Por el contrario, los clasificadores bayesianos han arrojado los peores resultados, exceptuando la red bayesiana entrenada con TAN. Por su parte, el KNN no ha conseguido resultados destacables para ninguno de los valores de K con los que se ha probado.

Tal y como ha ocurrido con la primera fase, el conjunto de características ha obtenido los mejores resultados para la mayoría los clasificadores empleados. Por este motivo, vamos a combinar este conjunto de características con cada uno de los grupos analizados hasta el momento.

- **Fase 2 - Características simples combinadas con las características del *Best Crossing Line Profile*.**

Tabla 4.14: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (AUC) empleando las características del *Best Crossing Line Profile* combinadas con las características simples para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
RB: K2	84,61	0,8111
RB: TAN	89,44	0,8714
Naïve Bayes	85,46	0,7733
SVM: Polinomial	90,62	0,9254
SVM: Polinomial norm.	86,59	0,9228
SVM: Pearson VII	87,09	0,8937
SVM: RBF	84,65	0,9102
KNN K=1	88,56	0,7664
KNN K=2	87,98	0,8273
KNN K=3	88,10	0,8493
KNN K=4	87,61	0,8635
KNN K=5	87,74	0,8716
AD: J48	89,68	0,8548
AD: Random Forest N=10	90,47	0,9486
AD: Random Forest N=25	91,03	0,9558
AD: Random Forest N=50	91,25	0,9581
AD: Random Forest N=75	91,42	0,9596
AD: Random Forest N=100	91,18	0,9597

Para este conjunto de características (véase la Tabla 4.14), formada por la combinación de características simples y las extraídas de la matriz del *Best Crossing Line Profile*, los clasificadores se han comportado, en general, con precisiones superiores al 88% y áreas bajo la curva ROC mayores de 0,90, en su mayoría. Concretamente, *Random Forest* configurado con 75 árboles es el clasificador que obtiene el resultado más destacado, junto con SVM con kernel polinomial. El clasificador Naïve Bayes, por su parte, ha obtenido los peores resultados para este conjunto de características. De igual modo, KNN no arroja unos resultados significativos para ninguno de los valores de K probados.

- **Fase 2 - Características simples combinadas con las características de la transformada rápida de Fourier.**

Tabla 4.15: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (*AUC*) empleando las características de la transformada rápida de Fourier combinadas con las características simples para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
<i>RB: K2</i>	84,44	0,8152
<i>RB: TAN</i>	89,42	0,8937
<i>Naïve Bayes</i>	86,36	0,7780
<i>SVM: Polinomial</i>	90,71	0,9395
<i>SVM: Polinomial norm.</i>	85,71	0,9249
<i>SVM: Pearson VII</i>	87,68	0,9009
<i>SVM: RBF</i>	82,93	0,9062
<i>KNN K=1</i>	90,05	0,7562
<i>KNN K=2</i>	89,18	0,7983
<i>KNN K=3</i>	88,26	0,8597
<i>KNN K=4</i>	87,56	0,8690
<i>KNN K=5</i>	87,95	0,8724
<i>AD: J48</i>	89,51	0,8591
<i>AD: Random Forest N=10</i>	90,39	0,9495
<i>AD: Random Forest N=25</i>	91,03	0,9588
<i>AD: Random Forest N=50</i>	90,99	0,9616
<i>AD: Random Forest N=75</i>	90,86	0,9617
<i>AD: Random Forest N=100</i>	90,84	0,9624

Con el grupo de características formado por la combinación de las características simples y las extraídas de la transformada rápida de Fourier (véase la Tabla 4.15), los clasificadores han obtenido, por norma general, unos valores cercanos al 90 % de precisión y áreas bajo la curva ROC superiores a 0,90. Particularmente, el clasificador *Random Forest*, con todas las configuraciones probadas, ha destacado sobre el resto de clasificadores con una precisión de más del 90 % y un área bajo la curva ROC superior a 0,94. Otro de los clasificadores destacados es SVM con kernel polinomial, que presenta una precisión del 90,71 % y un área bajo la curva ROC de 0,9395. El resultado más bajo sigue siendo el arrojado por el clasificador Naïve Bayes.

- **Fase 2 - Características simples combinadas con las características de la matriz de co-ocurrencia.**

Tabla 4.16: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (*AUC*) empleando las características de la matriz de co-ocurrencia combinadas con las características simples para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
<i>RB: K2</i>	84,07	0,8210
<i>RB: TAN</i>	89,14	0,8922
<i>Naïve Bayes</i>	85,93	0,7898
<i>SVM: Polinomial</i>	91,01	0,9467
<i>SVM: Polinomial norm.</i>	86,90	0,9309
<i>SVM: Pearson VII</i>	88,22	0,9070
<i>SVM: RBF</i>	83,56	0,9101
<i>KNN K=1</i>	89,57	0,7601
<i>KNN K=2</i>	89,12	0,8304
<i>KNN K=3</i>	89,29	0,8697
<i>KNN K=4</i>	88,64	0,8763
<i>KNN K=5</i>	88,43	0,8814
<i>AD: J48</i>	89,39	0,8640
<i>AD: Random Forest N=10</i>	90,41	0,9522
<i>AD: Random Forest N=25</i>	91,23	0,9597
<i>AD: Random Forest N=50</i>	91,44	0,9616
<i>AD: Random Forest N=75</i>	91,77	0,9635
<i>AD: Random Forest N=100</i>	91,96	0,9633

La Tabla 4.7 muestra los resultados obtenidos para el conjunto de características obtenidas a partir de la combinación de las características simples y de las extraídas a partir de la matriz de co-ocurrencia, donde los distintos clasificadores arrojan unas precisiones cercanas al 90% y unas áreas bajo la curva ROC superiores a 0,90. Profundizando, el resultado más destacable es el de *Random Forest* con todas las configuraciones probadas, así como SVM con kernel polinomial. Por contra, los clasificadores que peores resultados han obtenido para el grupo de características dado han sido los bayesianos a excepción de la red bayesiana entrenada con TAN. El clasificador KNN no arroja resultados significativos para ninguna de las configuraciones probadas.

• Fase 2 - Todas las características.

Tabla 4.17: Resultados obtenidos en términos de precisión (en porcentaje) y área bajo la curva ROC (*AUC*) empleando todas las características obtenidas para la Fase 2 (inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	AUC
<i>RB: K2</i>	84,63	0,8003
<i>RB: TAN</i>	89,53	0,8414
<i>Naïve Bayes</i>	85,35	0,7697
<i>SVM: Polinomial</i>	91,51	0,9343
<i>SVM: Polinomial norm.</i>	86,49	0,9208
<i>SVM: Pearson VII</i>	86,27	0,8763
<i>SVM: RBF</i>	84,10	0,9106
<i>KNN K=1</i>	88,99	0,7716
<i>KNN K=2</i>	88,32	0,8174
<i>KNN K=3</i>	88,00	0,8396
<i>KNN K=4</i>	87,76	0,8540
<i>KNN K=5</i>	87,52	0,8619
<i>AD: J48</i>	88,99	0,8583
<i>AD: Random Forest N=10</i>	89,93	0,9447
<i>AD: Random Forest N=25</i>	90,62	0,9532
<i>AD: Random Forest N=50</i>	90,84	0,9563
<i>AD: Random Forest N=75</i>	91,03	0,9560
<i>AD: Random Forest N=100</i>	91,10	0,9576

Para este conjunto de características (véase la Tabla 4.2), formada por la combinación de características simples y las extraídas de la matriz de co-ocurrencia, los clasificadores se han comportado, en general, con precisiones superiores al 92 % y áreas bajo la curva ROC mayores de 0,90, en su mayoría. Concretamente, SVM con kernel Pearson VII es el clasificador que obtiene el resultado más destacado, junto con *Random Forest* configurado con más de 25 árboles. El clasificador Naïve Bayes es el que peores resultados ha obtenido para este grupo de características junto con KNN.

4.3.3.2 Categorización directa

A diferencia del caso anterior, para esta categorización, tenemos discriminados los distintos defectos posibles en el conjunto de datos. A continuación, mostramos los resultados obtenidos empleando los mismos conjuntos de características que en la sección anterior.

- **Características extraídas del *Best Crossing Line Profile***

Tabla 4.18: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (*ATPR*), tasa agregada de falsos positivos (*AFPR*) y área bajo la curva ROC (*AUC*) empleando las características del *Best Crossing Line Profile* para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
<i>RB: K2</i>	80,13	0,7672	0,1962	0,8787
<i>RB: TAN</i>	93,33	0,2426	0,0108	0,8881
<i>Naïve Bayes</i>	61,31	0,9197	0,4086	0,8216
<i>SVM: Polinomial</i>	60,38	0,9092	0,4157	0,8441
<i>SVM: Polinomial norm.</i>	69,49	0,8645	0,3160	0,8650
<i>SVM: Pearson VII</i>	89,33	0,6367	0,0875	0,8975
<i>SVM: RBF</i>	60,15	0,9155	0,4189	0,8351
<i>KNN K=1</i>	89,67	0,5961	0,0806	0,7674
<i>KNN K=2</i>	91,28	0,5399	0,0585	0,8296
<i>KNN K=3</i>	88,56	0,6738	0,0987	0,8558
<i>KNN K=4</i>	89,43	0,6529	0,0876	0,8647
<i>KNN K=5</i>	87,45	0,7151	0,1137	0,8715
<i>AD: J48</i>	87,89	0,6137	0,1017	0,7735
<i>AD: Random Forest N=10</i>	91,41	0,6164	0,0634	0,9020
<i>AD: Random Forest N=25</i>	91,43	0,6422	0,0652	0,9159
<i>AD: Random Forest N=50</i>	91,75	0,6460	0,0621	0,9209
<i>AD: Random Forest N=75</i>	91,73	0,6478	0,0624	0,9224
<i>AD: Random Forest N=100</i>	91,78	0,6512	0,0622	0,9238

La Tabla 4.10 muestra los resultados obtenidos para el conjunto de características obtenidas a partir del *Best Crossing Line Profile*, donde los distintos clasificadores arrojan unas precisiones en torno al 80% y unas áreas bajo la curva ROC de cercanas a 0,85. Así, los resultados más destacables son los obtenidos por SVM con kernel Pearson VII y, por *Random Forest* configurado con 100 árboles. Por contra, el clasificador que peores resultados ha obtenido para el grupo de características dado ha sido KNN con todos los valores probados para la *K*.

• Características extraídas de la transformada rápida de Fourier

Tabla 4.19: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (ATPR), tasa agregada de falsos positivos (AFPR) y área bajo la curva ROC (AUC) empleando las características de la transformada rápida de Fourier para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
RB: K2	75,92	0,8422	0,2469	0,8715
RB: TAN	86,17	0,6784	0,1248	0,8821
Naïve Bayes	30,99	0,9774	0,7364	0,8306
SVM: Polinomial	67,80	0,8987	0,3333	0,8568
SVM: Polinomial norm.	68,82	0,8938	0,3225	0,8671
SVM: Pearson VII	75,89	0,8554	0,2469	0,8859
SVM: RBF	67,53	0,8993	0,3351	0,8499
KNN K=1	86,12	0,5702	0,1182	0,7553
KNN K=2	88,77	0,4922	0,0834	0,8224
KNN K=3	85,76	0,6631	0,1286	0,8519
KNN K=4	86,86	0,6353	0,1149	0,8635
KNN K=5	84,74	0,7159	0,1433	0,8694
AD: J48	85,98	0,5315	0,1165	0,7329
AD: Random Forest N=10	89,14	0,5420	0,0832	0,8754
AD: Random Forest N=25	89,30	0,5789	0,0840	0,8864
AD: Random Forest N=50	89,49	0,5840	0,0824	0,8907
AD: Random Forest N=75	89,49	0,5907	0,0829	0,8928
AD: Random Forest N=100	89,47	0,5876	0,0829	0,8942

Para este conjunto de características (véase la Tabla 4.19), obtenidas a partir de la transformada rápida de Fourier, los clasificadores se han comportado, en general, con precisiones por debajo del 90 % y áreas bajo la curva ROC cercanas a 0,90. Concretamente, *Random Forest* con todas las configuraciones probadas, es el clasificador que obtiene el resultado más destacado. De forma similar, el clasificador KNN con el valor de *K* igual a 5, arroja su mejor resultado con una precisión del 84,74 %, un ATPR de 0,7159, un AFPR de 0,1433 y un área bajo la curva ROC de 0,8694. Por el contrario, el clasificador Naïve Bayes es el que peores resultados ha obtenido para este grupo de características.

- Características extraídas de la matriz de co-ocurrencia

Tabla 4.20: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (ATPR), tasa agregada de falsos positivos (AFPR) y área bajo la curva ROC (AUC) empleando las características de la matriz de co-ocurrencia para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
RB: K2	61,27	0,6200	0,3877	0,7140
RB: TAN	79,67	0,4148	0,1758	0,7342
Naïve Bayes	42,76	0,8793	0,6000	0,7275
SVM: Polinomial	54,98	0,7301	0,4608	0,7628
SVM: Polinomial norm.	58,31	0,6704	0,4221	0,7594
SVM: Pearson VII	67,33	0,6129	0,3229	0,7742
SVM: RBF	53,48	0,7604	0,4781	0,7315
KNN K=1	79,18	0,4458	0,1836	0,6875
KNN K=2	84,08	0,3538	0,1233	0,7233
KNN K=3	77,92	0,4986	0,2011	0,7432
KNN K=4	80,05	0,4560	0,1751	0,7500
KNN K=5	76,28	0,5246	0,2204	0,7566
AD: J48	75,71	0,4374	0,2209	0,6628
AD: Random Forest N=10	83,23	0,4102	0,1375	0,7569
AD: Random Forest N=25	83,19	0,4294	0,1395	0,7697
AD: Random Forest N=50	83,39	0,4233	0,1368	0,7723
AD: Random Forest N=75	83,45	0,4250	0,1364	0,7739
AD: Random Forest N=100	83,49	0,4250	0,1361	0,7747

Con el grupo de características extraído de la matriz de co-ocurrencia (véase la Tabla 4.20), los clasificadores han obtenido, por norma general, unos valores cercanos al 80% de precisión y áreas bajo la curva ROC alrededor de 0,75. De forma particular, el clasificador *Random Forest* configurado con un número de árboles igual a 100, ha destacado sobre el resto de clasificadores con una precisión del 83,49%, un ATPR de 0,4250, un AFPR de 0,1361 y un área bajo la curva ROC de 0,7747. Otro de los clasificadores destacados es la red bayesiana entrenada con TAN, que presenta una precisión del 79,67%, un ATPR de 0,4148, un AFPR de 0,1758 y un área bajo la curva ROC de 0,7342. Por su parte, el clasificador Naïve Bayes ha arrojado los peores resultados para esta configuración.

- Características simples

Tabla 4.21: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (ATPR), tasa agregada de falsos positivos (AFPR) y área bajo la curva ROC (AUC) empleando las características simples para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
RB: K2	91,95	0,7283	0,0663	0,9171
RB: TAN	95,39	0,4479	0,0058	0,9185
Naïve Bayes	79,94	0,8772	0,2062	0,9089
SVM: Polinomial	90,13	0,8559	0,0954	0,9576
SVM: Polinomial norm.	92,62	0,8053	0,0649	0,9612
SVM: Pearson VII	96,47	0,8272	0,0248	0,9749
SVM: RBF	87,98	0,8562	0,1185	0,9528
KNN K=1	92,62	0,1960	0,0154	0,5983
KNN K=2	92,96	0,0993	0,0033	0,6342
KNN K=3	93,05	0,1314	0,0058	0,6741
KNN K=4	92,92	0,0900	0,0037	0,7088
KNN K=5	92,95	0,1123	0,0054	0,7381
AD: J48	93,89	0,7197	0,0444	0,8477
AD: Random Forest N=10	96,22	0,7591	0,0220	0,9626
AD: Random Forest N=25	96,46	0,7812	0,0212	0,9706
AD: Random Forest N=50	96,55	0,7815	0,0202	0,9743
AD: Random Forest N=75	96,60	0,7860	0,0200	0,9755
AD: Random Forest N=100	96,64	0,7849	0,0194	0,9763

La Tabla 4.21 muestra los resultados obtenidos para el conjunto de características que hemos denominado simples, donde los distintos clasificadores arrojan unas precisiones que oscilan entre el 90 % y el 95 % y unas áreas bajo la curva ROC de cercanas a 0,95. En concreto, los resultados más destacables son los obtenidos por SVM con kernel Pearson VII y, por *Random Forest* configurado con 25 árboles o más. Por su parte, entre los clasificadores bayesianos, la red bayesiana entrenada con TAN también obtiene un buen resultado. Por contra, el clasificador que peores resultados ha obtenido para el grupo de características dado ha sido KNN con todos los valores probados para la K .

Como ocurre con el enfoque anterior, los resultados que hemos obtenido con el conjunto de características simples son sustancialmente superiores a los que presentan el resto de conjuntos utilizados, por lo que las combinaremos entre sí para evaluar el mejor grupo de características a emplear.

- **Características simples combinadas con las características del *Best Crossing Line Profile***

Tabla 4.22: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (*ATPR*), tasa agregada de falsos positivos (*AFPR*) y área bajo la curva ROC (*AUC*) empleando las características del *Best Crossing Line Profile* combinadas con las características simples para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
<i>RB: K2</i>	92,30	0,6810	0,0586	0,8950
<i>RB: TAN</i>	94,13	0,2388	0,0018	0,8387
<i>Naïve Bayes</i>	78,45	0,8792	0,2222	0,8955
<i>SVM: Polinomial</i>	91,47	0,8278	0,0788	0,9589
<i>SVM: Polinomial norm.</i>	92,79	0,7866	0,0617	0,9601
<i>SVM: Pearson VII</i>	95,88	0,7891	0,0283	0,9674
<i>SVM: RBF</i>	90,20	0,8239	0,0923	0,9553
<i>KNN K=1</i>	92,57	0,2055	0,0170	0,6038
<i>KNN K=2</i>	92,91	0,1032	0,0044	0,6450
<i>KNN K=3</i>	92,97	0,1604	0,0090	0,6804
<i>KNN K=4</i>	92,97	0,1186	0,0058	0,7089
<i>KNN K=5</i>	93,00	0,1528	0,0083	0,7323
<i>AD: J48</i>	93,84	0,7034	0,0437	0,8396
<i>AD: Random Forest N=10</i>	96,25	0,7435	0,0205	0,9622
<i>AD: Random Forest N=25</i>	96,46	0,7661	0,0200	0,9699
<i>AD: Random Forest N=50</i>	96,59	0,7711	0,0189	0,9732
<i>AD: Random Forest N=75</i>	96,59	0,7719	0,0189	0,9744
<i>AD: Random Forest N=100</i>	96,61	0,7709	0,0186	0,9751

Para este conjunto de características (véase la Tabla 4.22), formada por la combinación de características simples y las extraídas del *Best Crossing Line Profile*, los clasificadores se han comportado, en general, con precisiones superiores al 92 % y áreas bajo la curva ROC mayores de 0,90. Concretamente, SVM con kernel Pearson VII es el clasificador que obtiene el resultado más destacado, junto con *Random Forest* configurado con más de 25 árboles. El clasificador Naïve Bayes es el que peores resultados ha obtenido para este grupo de características junto con KNN.

- **Características simples combinadas con las características de la transformada rápida de Fourier**

Tabla 4.23: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (ATPR), tasa agregada de falsos positivos (AFPR) y área bajo la curva ROC (AUC) empleando las características de la transformada rápida de Fourier combinadas con las características simples para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
RB: K2	91,08	0,7604	0,0779	0,9155
RB: TAN	95,07	0,3963	0,0049	0,8997
Naïve Bayes	78,28	0,8949	0,2253	0,9094
SVM: Polinomial	90,72	0,8504	0,0886	0,9574
SVM: Polinomial norm.	92,59	0,8035	0,0655	0,9611
SVM: Pearson VII	96,41	0,8283	0,0255	0,9762
SVM: RBF	88,67	0,8522	0,1109	0,9529
KNN K=1	92,80	0,1895	0,0128	0,5951
KNN K=2	92,96	0,0998	0,0033	0,6299
KNN K=3	92,96	0,1274	0,0064	0,6681
KNN K=4	92,90	0,0924	0,0040	0,7039
KNN K=5	92,89	0,1098	0,0059	0,7355
AD: J48	93,88	0,7218	0,0446	0,8411
AD: Random Forest N=10	96,28	0,7553	0,0212	0,9623
AD: Random Forest N=25	96,51	0,7766	0,0203	0,9708
AD: Random Forest N=50	96,62	0,7786	0,0191	0,9741
AD: Random Forest N=75	96,68	0,7852	0,0190	0,9752
AD: Random Forest N=100	96,73	0,7859	0,0185	0,9763

Con el grupo de características formado por la combinación de las características simples y las extraídas de la transformada rápida de Fourier (véase la Tabla 4.23), los clasificadores han obtenido, por norma general, unos valores de más del 95 % de precisión y áreas bajo la curva ROC mayores de 0,95. Particularmente, el clasificador *Random Forest* configurado con un número de árboles de superior a 25, ha destacado sobre el resto de clasificadores con una precisión superior al 96 %, un ATPR de 0,77, un AFPR de tan sólo un 0,02 y un área bajo la curva ROC de 0,97. Otro de los clasificadores destacados es SVM con kernel Pearson VII, que presenta una precisión del 96,41 %, un ATPR de 0,8283, un AFPR de 0,0255 y un área bajo la curva ROC de 0,9762. El clasificador Naïve Bayes ha arrojado los peores resultados para esta configuración. Asimismo, KNN no presenta unos resultados significativos para este conjunto de características.

- **Características simples combinadas con las características de la matriz de co-ocurrencia**

Tabla 4.24: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (*ATPR*), tasa agregada de falsos positivos (*AFPR*) y área bajo la curva ROC (*AUC*) empleando las características de la matriz de co-ocurrencia combinadas con las características simples para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
<i>RB: K2</i>	91,00	0,7106	0,0755	0,9012
<i>RB: TAN</i>	95,08	0,3915	0,0044	0,9012
<i>Naïve Bayes</i>	80,11	0,8779	0,2043	0,9048
<i>SVM: Polinomial</i>	91,23	0,8380	0,0822	0,9592
<i>SVM: Polinomial norm.</i>	93,06	0,7946	0,0593	0,9622
<i>SVM: Pearson VII</i>	96,42	0,8250	0,0252	0,9774
<i>SVM: RBF</i>	89,20	0,8477	0,1048	0,9557
<i>KNN K=1</i>	92,66	0,2025	0,0154	0,6002
<i>KNN K=2</i>	92,92	0,0974	0,0036	0,6345
<i>KNN K=3</i>	93,01	0,1359	0,0065	0,6769
<i>KNN K=4</i>	92,90	0,0915	0,0040	0,7134
<i>KNN K=5</i>	92,97	0,1185	0,0055	0,7462
<i>AD: J48</i>	93,96	0,7192	0,0436	0,8411
<i>AD: Random Forest N=10</i>	96,14	0,7521	0,0224	0,9618
<i>AD: Random Forest N=25</i>	96,46	0,7818	0,0212	0,9711
<i>AD: Random Forest N=50</i>	96,57	0,7812	0,0200	0,9741
<i>AD: Random Forest N=75</i>	96,57	0,7831	0,0201	0,9751
<i>AD: Random Forest N=100</i>	96,61	0,7844	0,0197	0,9760

La Tabla 4.24 muestra los resultados obtenidos para el conjunto de características obtenidas a partir de la combinación de las características simples y las obtenidas de la matriz de co-ocurrencia, donde los distintos clasificadores arrojan unas precisiones en torno al 95 % y unas áreas bajo la curva ROC de cercanas a 0,95. Profundizando, los resultados más destacables son los obtenidos por SVM con kernel Pearson VII y, por *Random Forest* configurado con más de 25 árboles. Por su parte, entre los clasificadores bayesianos, la red bayesiana entrenada con TAN también arroja un buen resultado. Por contra, el clasificador que peores resultados ha obtenido para el grupo de características dado ha sido Naïve Bayes.

• Todas las características

Tabla 4.25: Resultados obtenidos en términos de precisión (en porcentaje), tasa agregada de verdaderos positivos (*ATPR*), tasa agregada de falsos positivos (*AFPR*) y área bajo la curva ROC (*AUC*) empleando todas las características obtenidas para la categorización directa (tolerable / inclusión / unión fría / falta de llenado).

Algoritmo	Prec.(%)	ATPR	AFPR	AUC
<i>RB: K2</i>	90,88	0,6990	0,0754	0,8848
<i>RB: TAN</i>	93,80	0,1926	0,0014	0,7983
<i>Naïve Bayes</i>	77,61	0,8937	0,2322	0,8953
<i>SVM: Polinomial</i>	92,53	0,8152	0,0665	0,9593
<i>SVM: Polinomial norm.</i>	93,12	0,7863	0,0581	0,9606
<i>SVM: Pearson VII</i>	95,81	0,7883	0,0289	0,9680
<i>SVM: RBF</i>	91,11	0,8127	0,0818	0,9560
<i>KNN K=1</i>	92,60	0,1921	0,0157	0,5981
<i>KNN K=2</i>	92,94	0,1071	0,0048	0,6460
<i>KNN K=3</i>	92,98	0,1649	0,0094	0,6838
<i>KNN K=4</i>	92,95	0,1176	0,0060	0,7139
<i>KNN K=5</i>	93,00	0,1501	0,0081	0,7381
<i>AD: J48</i>	93,51	0,6930	0,0467	0,8284
<i>AD: Random Forest N=10</i>	96,13	0,7359	0,0213	0,9614
<i>AD: Random Forest N=25</i>	96,40	0,7597	0,0201	0,9702
<i>AD: Random Forest N=50</i>	96,54	0,7655	0,0190	0,9737
<i>AD: Random Forest N=75</i>	96,59	0,7699	0,0188	0,9747
<i>AD: Random Forest N=100</i>	96,59	0,7692	0,0187	0,9752

Para este conjunto de características (véase la Tabla 4.25), formada por la combinación de características simples y las extraídas del *Best Crossing Line Profile*, los clasificadores se han comportado, en general, con precisiones superiores al 92% y áreas bajo la curva ROC mayores de 0,95, en su mayoría. Concretamente, SVM con kernel Pearson VII es el clasificador que obtiene el resultado más destacado, junto con *Random Forest* configurado con más de 50 árboles. El clasificador Naïve Bayes es el que peores resultados ha obtenido para este grupo de características junto con KNN, que no muestra ningún resultado aceptable con ninguna de sus configuraciones.

4.4 Discusión de los resultados

Los resultados obtenidos mediante las distintas configuraciones planteadas demuestran que es posible detectar y categorizar defectos superficiales en piezas

de fundición de hierro nodular. En general, los mejores resultados son los que han arrojado SVM con kernel Pearson VII y *Random Forest* configurado con 100 árboles.

Tabla 4.26: Comparación de los mejores clasificadores en términos de precisión, TPR, FPR y área bajo la curva ROC para cada conjunto de características empleadas para la fase 1 de la categorización en dos fases, donde (1) se corresponde con el conjunto de características simples, (2) con la combinación de las características simples y las extraídas del *Best Crossing Line Profile*, (3) con la combinación de las características simples y las extraídas de la transformada rápida de Fourier, (4) con la combinación de las características simples y las extraídas de la matriz de co-ocurrencia, y por último, (5) con la combinación de todas las características.

Algoritmo	Medida	(1)	(2)	(3)	(4)	(5)
SVM: Pearson VII	Prec.(%)	97,27	96,85	97,31	97,29	96,88
	TPR	0,8164	0,7793	0,8146	0,8196	0,7789
	FPR	0,0273	0,0315	0,0269	0,0271	0,0312
	AUC	0,9742	0,9726	0,9770	0,9770	0,9752
AD: Random Forest N=100	Prec.(%)	97,06	97,09	97,10	97,02	97,03
	TPR	0,7932	0,7858	0,7939	0,7922	0,7806
	FPR	0,7932	0,0291	0,0290	0,0298	0,0297
	AUC	0,9744	0,9746	0,9741	0,9749	0,9740

Como podemos observar en la Tabla 4.26, aunque los resultados son muy similares entre si, destacan los valores arrojados por SVM con un kernel Pearson VII para el conjunto de características formado por la combinación de las simples y las extraídas de la matriz de co-ocurrencia, con una precisión del 97,29%, un TPR de 0,8196, un FPR de 0,271 y un área bajo la curva ROC de 0,9770. Este resultado demuestra la validez del método presentando para determinar si una región es defectuosa o no, sin entrar en el tipo de defecto que pueda existir.

Tabla 4.27: Comparación de los mejores clasificadores en términos de precisión y área bajo la curva ROC para cada conjunto de características empleadas para la fase 2 de la categorización en dos fases, donde (1) se corresponde con el conjunto de características simples, (2) con la combinación de las características simples y las extraídas del *Best Crossing Line Profile*, (3) con la combinación de las características simples y las extraídas de la transformada rápida de Fourier, (4) con la combinación de las características simples y las extraídas de la matriz de co-ocurrencia, y por último, (5) con la combinación de todas las características.

Algoritmo	Medida	(1)	(2)	(3)	(4)	(5)
SVM: Pearson VII	Prec.(%)	88,22	87,09	87,68	88,22	86,27
	AUC	0,9097	0,8937	0,9009	0,9070	0,8763
AD: Random Forest N=100	Prec.(%)	91,30	91,18	90,84	91,96	91,10
	AUC	0,9638	0,9597	0,9624	0,9633	0,9576

En cuanto a la segunda etapa de la categorización en dos fases, la Tabla 4.27 muestra los resultados que han obtenido los mejores clasificadores para

la determinar el tipo de defecto existente. A diferencia de lo que ocurría en la fase 1, el mejor resultado es el de *Random Forest* con un número de árboles igual a 100, y con el conjunto de características simples combinadas con las extraídas de la matriz de co-ocurrencia. En concreto, la precisión alcanza el 91,96% y el área bajo la curva ROC es de 0,9633.

Tabla 4.28: Comparación de los mejores clasificadores en términos de precisión, ATPR, AFPR y área bajo la curva ROC para cada conjunto de características empleadas para la categorización directa, donde (1) se corresponde con el conjunto de características simples, (2) con la combinación de las características simples y las extraídas del *Best Crossing Line Profile*, (3) con la combinación de las características simples y las extraídas de la transformada rápida de Fourier, (4) con la combinación de las características simples y las extraídas de la matriz de co-ocurrencia, y por último, (5) con la combinación de todas las características.

Algoritmo	Medida	(1)	(2)	(3)	(4)	(5)
SVM: Pearson VII	Prec.(%)	96,47	95,88	96,41	96,42	95,81
	ATPR	0,8272	0,7891	0,8283	0,8250	0,7883
	AFPR	0,0248	0,0283	0,0255	0,0252	0,0289
	AUC	0,9749	0,9674	0,9762	0,9774	0,9680
AD: Random Forest N=100	Prec.(%)	96,64	96,61	96,73	96,61	96,59
	ATPR	0,7849	0,7709	0,7859	0,7844	0,7692
	AFPR	0,019	0,0186	0,0185	0,0197	0,0187
	AUC	0,9763	0,9751	0,9763	0,9760	0,9752

En la Tabla 4.28 podemos observar los clasificadores con un mejor comportamiento para la categorización directa. En concreto, los resultados de *Random Forest* y de SVM no muestran apenas variaciones en cuanto a precisión y área bajo la curva ROC se refiere. La diferencia principal entre ambos se encuentra en los valores para el ATPR y para el AFPR. En el caso de SVM, el ATPR es mayor, indicando de esta forma, que acierta un mayor número de regiones al categorizarlas como uno de los 3 tipos de defecto estudiados. Por su parte, *Random Forest* categoriza de una forma más precisa las regiones tolerables. En cuanto al conjunto de características, destacamos los resultados obtenidos por la combinación de las características simples y las extraídas de la matriz de co-ocurrencia.

Si comparamos los resultados obtenidos mediante la categorización en dos fases y, la categorización directa, observamos que la cantidad de defectos detectados es muy similar en ambos casos, siendo de 0,8196 en el primer caso, y de 0,8283 en el segundo.

A la vista de los resultados finales obtenidos, podemos afirmar que ambos enfoques son totalmente válidos para trabajar en un entorno de fundición. Para ello, habrá que realizar un estudio en cada caso, que determinará cual de los dos enfoques emplear, dependiendo generalmente del grado de trazabilidad que sea necesario.

4.5 Sumario

En este capítulo hemos analizado el uso de diferentes conjuntos de características para la categorización de defectos superficiales en piezas de fundición de hierro nodular. Para ello: (i) hemos extraído distintos tipos de características a partir de las regiones segmentadas en el anterior capítulo y (ii) empleando algoritmos de aprendizaje automático supervisado, las hemos categorizado en base a tres tipos de defecto estudiados.

En primer lugar, hemos partido de las regiones marcadas por el método de segmentación descrito en el Capítulo 3. De cada una de ellas hemos extraído diferentes grupos de características, formando de esta forma, 8 conjuntos de datos diferentes.

Posteriormente, aplicando algoritmos de aprendizaje automático supervisado, hemos categorizado las regiones en los diferentes defectos definidos. Para ello hemos planteado dos enfoques de categorización: (i) la basada en dos fases y (ii) la directa. En la primera de ellas hemos dividido la clasificación en dos fases, determinando en primer lugar si la región es tolerable o no y, en la segunda de ellas, el tipo de defecto que contiene en caso de que haya sido clasificada como defectuosa. Por contra, en el segundo enfoque, se determina si la región es tolerable o si contiene alguno de los defecto estudiados.

Los resultados que hemos obtenido superan, en su gran mayoría, el 96 % de precisión, con lo que podemos afirmar que el sistema propuesto se presenta como una alternativa viable para la inspección visual de piezas de fundición de hierro nodular.

«Observa las flores del cerezo, podrías dedicar tu vida entera a la búsqueda de una sólo y no habrías desperdiciado tu tiempo. Perfectas, simplemente perfectas.»

Katsumoto (Personaje de ficción)

CAPÍTULO

5

Optimización del método propuesto

A través de la presente tesis, hemos planteado, diseñado, implementado y evaluado un nuevo método para la detección y categorización de defectos superficiales en piezas de fundición de hierro nodular. En este sentido, y aunque los resultados obtenidos confirman la validez del método propuesto, hemos continuado buscando debilidades del sistema, a fin de optimizarlo. Por este motivo, en este capítulo planteamos mejoras para las distintas fases del método propuesto: (i) aplicadas al método de segmentación, para reducir el número de modelos contra los que comparar; y (ii) aplicadas a los algoritmos de aprendizaje automático supervisado, ya que éstos requieren un alto número de instancias etiquetadas para funcionar de forma correcta.

El resto del capítulo sigue de la siguiente forma. La Sección 5.1 presenta de forma detallada la mejora que planteamos para el método de segmentación. La Sección 5.2 presenta la mejora que proponemos para el método de categorización. La Sección 5.3 expone los resultados obtenidos a través de la experimentación realizada para las mejoras de los métodos de segmentación y categorización. Por último, la Sección 5.4 resume las aportaciones realizadas en este capítulo.

5.1 Mejora propuesta para el método de segmentación

En el Capítulo 3, hemos propuesto un método de segmentación de imágenes basado en el mapa de vectores normales de la superficie codificado en RGB.

Para evaluar el método propuesto, hemos empleado el porcentaje total de defectos que es capaz de detectar. En concreto, este valor ha alcanzado el 59,70 %. En este sentido, destacamos que este porcentaje es dependiente de la dirección y sentido en el que el dispositivo de captura ha obtenido los datos.

Por otro lado, este método de segmentación utiliza las piezas que han sido aceptadas como modelos contra los que comparar las superficies que queremos evaluar. En nuestro caso concreto, el conjunto de modelos está formado por 236 ejemplares, lo que conlleva 236 comparaciones distintas para cada pieza que queremos examinar.

Por este motivo, la primera mejora que proponemos es una reducción del conjunto de modelos, disminuyendo el número de comparaciones y por ende, el coste computacional. En concreto, proponemos una reducción basada en un algoritmo de *clustering* o agrupación.

5.1.1 Algoritmos de *clustering*

Los algoritmos de agrupación o *clustering* se caracterizan por dividir un conjunto de datos en grupos significativos que se forman en base a la cercanía entre los elementos. Para ello, se calculan las distancias existentes entre cada una de las instancias del conjunto [Kum00].

Los tipos de *clustering* más comunes son: (i) el jerárquico y (ii) el particional. El primero de ellos genera los grupos de forma anidada, estableciendo de esta forma una relación jerárquica entre los diferentes *clusters*. Por el contrario, todos los grupos que genera el *clustering* particional se encuentran en un único nivel [Kum00].

Los algoritmos particionales, a su vez, se pueden dividir en dos tipos: (i) aquellos en los que se especifica el número de conjuntos o grupos a generar, por ejemplo *K-means*; y (ii) los que generan los *clusters* en base a un umbral que determina la distancia máxima entre los dos miembros del grupo más alejados entre sí, siendo uno de los más empleados el *Quality Threshold clustering*. En nuestro caso, hemos escogido este último ya que, *a priori*, desconocemos el número de grupos que necesitamos generar.

5.1.1.1 El algoritmo *Quality Threshold*

El algoritmo QT fue propuesto por Heyer et ál. [HKY99] para extraer la información relevante contenida en conjuntos de datos de gran tamaño. En concre-

to, hemos utilizado la implementación del algoritmo QT propuesta por Ugarte-Pedrero et ál. [UPSGB11], que en su caso, se planteó para reducir el volumen de su conjunto de datos en un problema de clasificación basado en anomalías.

En la Figura 5.1 podemos ver el pseudo-código de la implementación del algoritmo de *clustering Quality Threshold* de estos autores, donde se explica cómo se generan los vectores centroe, que son los que representan a cada uno de los *clusters* creados.

```

entrada: El conjunto de datos original  $\mathcal{V}$ , el umbral para cada grupo umbral, y el número
           mínimo de vectores para un grupo minNumVectores
salida : El conjunto de datos reducido  $\mathcal{R}$ 

// Calcula la distancia entre un vector y el resto de vectores del
// conjunto de datos.
foreach  $\{v_i | v_i \in \mathcal{V}\}$  do
    foreach  $\{v_j | v_j \in \mathcal{V}\}$  do
        // Si la distancia entre un vector  $v_j$  y otro  $v_i$  es menor que
        // el umbral especificado, añadimos el vector  $v_j$  al cluster
        // potencial  $\mathcal{A}_i$ , asociado al vector  $v_i$ 
        if distancia( $v_i, v_j$ )  $\geq$  umbral then
             $\mathcal{A}_i$ .añadir( $v_j$ )
        end
    end
end

// En cada vuelta se selecciona el grupo con mayor número de
// vectores.
while  $\exists \mathcal{A}_i \in \mathcal{A} : |\mathcal{A}_i| \geq \text{minNumVectores}$  and  $\forall \mathcal{A}_j \in \mathcal{A} : |\mathcal{A}_i| \geq |\mathcal{A}_j|$  and  $i \neq j$  do
    // Añadir el vector centroe que representa al cluster al
    // conjunto de datos reducido final
     $\mathcal{R}$ .añadir(centroe( $\mathcal{A}_i$ ))
    // Descartamos los clusters potenciales asociados al vector
     $v_j \in \mathcal{A}_i$ 
    foreach  $\{v_j | v_j \in \mathcal{A}_i\}$  do
         $\mathcal{A}$ .eliminar( $\mathcal{A}_j$ )  $\mathcal{V}$ .eliminar( $v_j$ )
    end
    // Eliminamos los vectores  $v_j \in \mathcal{A}_i$  de los clusters  $\mathcal{A}_k$  que queden
    // en  $\mathcal{A}$ 
    foreach  $\{\mathcal{A}_k | \mathcal{A}_k \in \mathcal{A}\}$  do
        foreach  $\{v_j | v_j \in \mathcal{A}_k \text{ and } v_j \in \mathcal{A}_i\}$  do
             $\mathcal{A}_k$ .eliminar( $v_j$ )
        end
    end
end

// Añadimos los vectores que quedan al conjuntos de datos reducido
// final
foreach  $\{v_j | v_j \in \mathcal{V}\}$  do
     $\mathcal{R}$ .añadir( $v_j$ )
end

```

Figura 5.1: Pseudo-código de la implementación del algoritmo QT propuesta por Ugarte-Pedrero et ál. [UPSGB11].

No obstante, cabe destacar que, al ser un algoritmo que genera los grupos en base a un umbral dado, desconocemos el número de *clusters* distintos que va a crear *a priori*. Por esta razón, para realizar una experimentación lo más exhaustiva posible, hemos asignado distintos valores al umbral y hemos examinado cómo influye, tanto en la generación de centroides como en el sistema final.

Como podemos observar en el algoritmo de la Figura 5.1, el conjunto de datos de entrada está formado por un grupo de vectores. De esta manera, hemos transformado las matrices de alturas de las piezas modelo en vectores. A continuación, mostramos el proceso que hemos seguido.

En primer lugar, y partiendo de la matriz $\mathcal{A}_{m \times n}$ (véase Ecuación 5.1):

$$\mathcal{A}_{m \times n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \\ & & \dots & & \\ a_{m-1,1} & a_{m-1,2} & \dots & a_{m-1,n-1} & a_{m-1,n} \\ a_{m,1} & a_{m,2} & \dots & a_{m,n-1} & a_{m,n} \end{pmatrix} \quad (5.1)$$

donde cada $a_{x,y}$ representa la altura en cada uno de los puntos del espacio (x, y) , generamos el vector $\vec{v}$. Como resultado, hemos ordenado todos los elementos de $\mathcal{A}_{m \times n}$ en una sola fila, tal y como aparece en la Ecuación 5.2.

$$\vec{v} = \{a_{1,1}, a_{1,2}, \dots, a_{1,n-1}, a_{1,n}, a_{2,1}, a_{2,2}, \dots, a_{m,n-1}, a_{m,n}\} \quad (5.2)$$

5.1.2 Validación empírica de la mejora propuesta para el método de segmentación

En esta sección, aplicamos la mejora propuesta y realizamos un análisis completo del impacto que ésta supone sobre el resto del proceso. Por este motivo, comenzamos con la especificación de la metodología de experimentación que seguimos. Posteriormente, continuamos con la evaluación de los resultados obtenidos en la experimentación.

5.1.2.1 Metodología de evaluación y medición de los resultados

Para evaluar los resultados obtenidos con esta mejora, vamos a realizar los siguientes dos pasos: (i) en el primero, examinamos el nuevo porcentaje de defectos detectados por el método de segmentación; (ii) en el segundo, evaluamos el impacto que produce el nuevo método de segmentación en la posterior categorización de defectos.

1. **Método de segmentación.** Para evaluar el nuevo método de segmentación, vamos a emplear la misma métrica que hemos utilizado en la

Sección 3.6.1. Concretamente, obtenemos el porcentaje de defectos que hemos sido capaces de detectar con el nuevo método.

En cuanto al algoritmo de *clustering*, hemos establecido una serie de valores para el umbral de similitud, y así, determinar cual es el óptimo. En concreto, hemos realizado la experimentación con los siguientes valores: 300, 325, 350, 375, 400, 425, 450, 475 y 500. En este sentido, el límite inferior viene determinado por el primer umbral para el que se reduce el conjunto de modelos, mientras que el último, se establece en base al valor a partir del cual, el método de segmentación comienza a perder precisión.

2. **Método de categorización.** Para evaluar el impacto que supone el nuevo método de segmentación en el resto del proceso, y partiendo de los datos extraídos del nuevo conjunto de regiones detectadas, hemos lanzado de nuevo la mejor configuración obtenida en la Sección 4.4 para la categorización de defectos.

En cuanto a las métricas empleadas, nos hemos basado en las ya descritas en la Sección 4.3.2. Éstas son: (i) la tasa de verdaderos positivos (TPR), (ii) la tasa de falsos positivos (FPR), (iii) la tasa agregada de verdaderos positivos (ATPR), (iv) la tasa agregada de falsos positivos (AFPR), (v) la precisión, y por último, (vi) el área bajo la curva ROC (AUC).

Por otra parte, y debido a que el conjunto de datos obtenido para la categorización está más desbalanceado que el utilizado en el Capítulo 4, hemos probado a utilizar diferentes estrategias para balancear el conjunto de datos. siendo éstas: (i) SMOTE, (ii) *Resample* y (iii) *Spread Subsample*.

Cabe destacar que, tanto SMOTE como *Resample*, son técnicas que afectan a la clase minoritaria, generando muestras sintéticas para balancear el conjunto de datos de aprendizaje. Por el contrario, *Spread Subsample* elimina muestras de la clase mayoritaria para igualar de esta forma la distribución de clases en el conjunto de datos. Estos tres algoritmos se encuentran implementados dentro de la herramienta de aprendizaje automático WEKA.

5.1.2.2 Resultados obtenidos mediante la experimentación

La primera evaluación de resultados que vamos a hacer es la del método de segmentación. Para ello, hemos generado los vectores centroides para cada uno de los umbrales definidos.

Como podemos observar en la Tabla 5.1, el porcentaje de defectos es directamente proporcional al umbral utilizado. Asimismo, es inversamente proporcional al número de vectores centroe generados. En este sentido, el principal

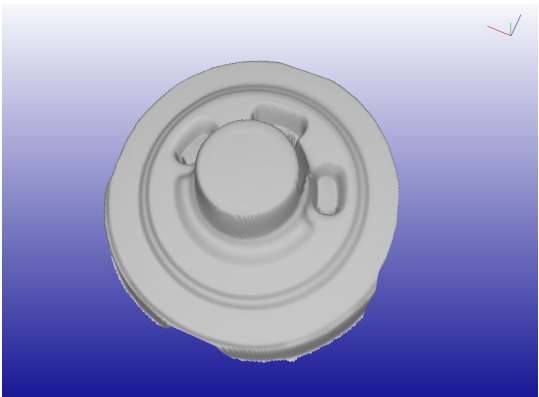


Figura 5.2: Visualización de la matriz de alturas de un modelo centroeide mediante el renderizado de una malla 3D.

Tabla 5.1: Resultados obtenidos en cuanto a número de defectos detectados por el método de segmentación para cada umbral de similitud probado.

Umbral	Centroides generados	Defectos detectados(%)
300	236	59,71
325	234	59,87
350	218	60,69
375	182	62,97
400	128	66,72
425	68	77,00
450	51	78,79
475	30	87,93
500	18	94,13

inconveniente que hemos observado, es que cuantos menos centroides modelo tenemos para comparar en el método de segmentación, las regiones potencialmente defectuosas son más grandes y menos precisas. Esto, se explica por el propio algoritmo de segmentación, y más concretamente, por el paso en el que obtenemos la intersección de todas las diferencias contra los modelos. Otra de las razones que influyen, es que los modelos centroides presentan una superficie con menor rugosidad que en un modelo real (véase la Figura 5.2).

En la Figura 5.3 podemos ver la evolución del resultado del método de segmentación para un ejemplo dado, de donde obtenemos la conclusión, de que cuanto más incrementamos el umbral de similitud del algoritmo de *clustering*, aparecen un número mayor de regiones. De esta forma, aumentamos no sólo los defectos, sino que también hacemos lo propio con zonas de la superficie que presentan alguna estructura regular. Por otra parte comprobamos, que el

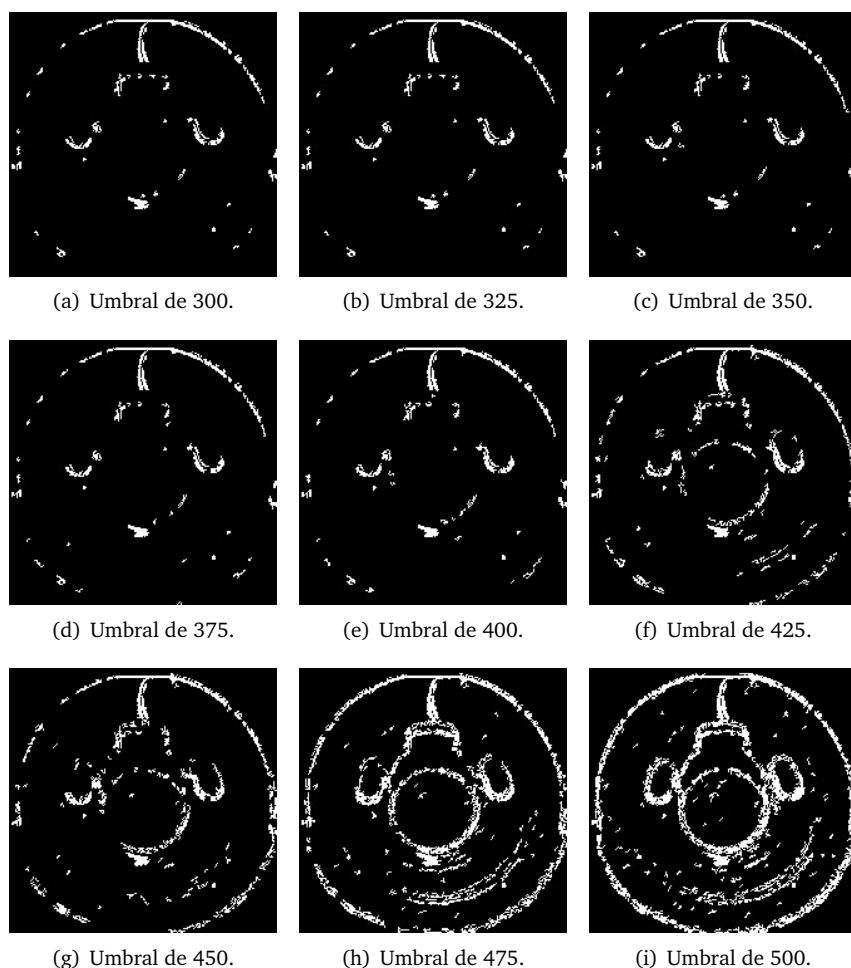


Figura 5.3: Ejemplo del resultado de segmentación de una pieza defectuosa con los distintos umbrales probados.

tamaño de las regiones crece de forma directamente proporcional al umbral, disminuyendo la precisión a la hora de marcar la forma y el tamaño real del defecto.

Por estas razones, a la hora de escoger el umbral óptimo hemos tenido en cuenta, además del porcentaje de defectos detectados, el tamaño de las regiones que los delimitan. Basándonos en estos conceptos, el valor que hemos escogido para el umbral del algoritmo de *clustering* es de 450. Con él, hemos obtenido 51 centroides, así como un porcentaje de defectos detectados del 78,79 %. Este valor aumenta con respecto al método de segmentación descrito

en la Sección 3.6.1 en 19,09 puntos porcentuales, o lo que es lo mismo, un incremento del 31,83 % en la detección de defectos.

En definitiva, hemos obtenido dos mejoras significativas en el método de segmentación: (i) disminuir el número modelos y como consecuencia, el tiempo empleado en la comparación de las imágenes contra los modelos, y (ii) aumentar el número de defectos detectados.

A continuación, hemos extraído las características simples combinadas con las obtenidas a partir de la matriz de co-ocurrencia de cada una de las regiones detectadas, y hemos generado un nuevo conjunto de datos. En la Tabla 5.2, presentamos la distribución de segmentos que hemos obtenido para cada uno de los tipos de defecto estudiados.

Tabla 5.2: Número de regiones con cada tipo de defecto.

Tipo	Número de segmentos
<i>Tolerable</i>	33216
<i>Inclusión</i>	553
<i>Unión fría</i>	20
<i>Falta de llenado</i>	60

Tal y como podemos observar en la Tabla 5.2, el conjunto de datos con el que vamos a trabajar se encuentra muy desbalanceado. Debido a ello, aplicamos distintas estrategias para que esta distribución de clases no influya de forma negativa en la categorización de defectos (véase la Sección 5.1.2.1).

Una vez obtenido el conjunto de datos para la categorización, hemos aplicado dos estrategias de resolución: (i) la categorización en dos fases, y (ii) la categorización directa.

Para la categorización en dos fases, comenzamos con una primera clasificación binaria (tolerable o defecto), y posteriormente, eliminando las regiones tolerables, categorizamos en base al defecto que pueda contener (inclusión, unión fría o falta de llenado).

Tabla 5.3: Resultados obtenidos para la fase 1 de la categorización en dos fases empleando el clasificador Random Forest configurado con 100 árboles, y utilizando la combinación de las características simples y las extraídas de la matriz de co-ocurrencia en términos de precisión, TPR, FPR y área bajo la curva ROC.

Preprocesado	Precisión(%)	TPR	FPR	AUC
<i>SMOTE</i>	98,73	0,6098	0,0055	0,9632
<i>Resample</i>	98,92	0,6223	0,0038	0,9704
<i>Spread Subsample</i>	90,75	0,8988	0,0923	0,9657

En la Tabla 5.3, mostramos los resultados obtenidos para la primera fase de la categorización en dos fases. A la vista de las cifras obtenidas, observa-

mos que *Spread Subsample* es el que mejor comportamiento presenta, con una precisión del 90,75 %, un TPR de 0,8988, un FPR de 0,0923 y un área bajo la curva ROC de 0,9657. Si bien es verdad que la precisión es inferior a la mostrada por las otras dos técnicas de balanceado, la cantidad de regiones defectuosas que conseguimos detectar es superior, sin penalizar en exceso los falsos positivos.

Respecto a la segunda fase de la categorización en dos fases, los resultados son los que arroja la Tabla 5.4. En este caso, el mejor resultado se obtiene aplicando SMOTE al conjunto de datos de entrenamiento, con una precisión del 92,70 % y un área bajo la curva ROC de 0,9612.

Tabla 5.4: Resultados obtenidos para la fase 2 de la categorización en dos fases empleando el clasificador Random Forest configurado con 100 árboles, y utilizando la combinación de las características simples y las extraídas de la matriz de co-ocurrencia en términos de precisión y área bajo la curva ROC.

Preprocesado	Precisión(%)	AUC
SMOTE	92,70	0,9612
Resample	89,92	0,9531
Spread Subsample	84,61	0,9225

En cuanto a la categorización directa, en la Tabla 5.5, mostramos los resultados obtenidos al aplicar la categorización directa. El resultado más destacable es el obtenido con la técnica *Resample* con una precisión del 98,77 %, un ATPR de 0,6814, un AFPR de 0,0061 y un área bajo la curva ROC de 0,9747.

Tabla 5.5: Resultados obtenidos para la categorización directa empleando el clasificador *Random Forest* configurado con 100 árboles, y utilizando la combinación de las características simples y las extraídas de la matriz de co-ocurrencia en términos de precisión, ATPR, AFPR y área bajo la curva ROC.

Preprocesado	Precisión(%)	ATPR	AFPR	AUC
SMOTE	98,73	0,6716	0,0062	0,9694
Resample	98,77	0,6814	0,0061	0,9747
Spread Subsample	70,79	0,9267	0,2929	0,9133

5.2 Mejora propuesta para el método de categorización

Otra de las mejoras que proponemos se aplica al método de categorización. Como hemos observado en el Capítulo 4, la estrategia seguida para la categorización de defectos se basa en el uso de algoritmos de aprendizaje automático

supervisado. Estas técnicas se caracterizan por tener resultados dependientes del volumen de datos utilizado para el aprendizaje, o dicho con otras palabras, a mayor cantidad de datos empleados en el aprendizaje, el modelo generado tendrá una mayor tasa de acierto. El principal problema asociado a este concepto es el esfuerzo que supone etiquetar toda esta cantidad de información.

En este contexto, durante los últimos años han aparecido diversos enfoques orientados a la reducción del esfuerzo de etiquetado. Uno de los enfoques más comunes es el aprendizaje automático semi-supervisado, que se ha aplicado en áreas como el filtrado de correo electrónico no deseado [Pfa06, SQ10, LSS⁺12], o el reconocimiento de imagen [WKC10, SFF10, FD11], mostrando en todos estos casos una reducción considerable de los datos etiquetados.

Concretamente, para mejorar nuestro método de categorización, nos centramos en los clasificadores colectivos, que componen una rama aprendizaje semi-supervisado. Este tipo de algoritmos son conocidos en áreas como la detección de *spam* [LSS⁺12], o la detección de malware empaquetado [SUPS⁺11]. Este enfoque, optimiza la categorización de defectos minimizando el esfuerzo empleado en la fase de etiquetado, y todo ello sin comprometer la precisión del mismo. En la siguiente sección analizaremos en qué consiste, así como los distintos algoritmos que hemos aplicado como mejora al sistema propuesto.

5.2.1 Clasificación colectiva

La clasificación colectiva se basa en un problema de optimización combinatoria, en la que dada una un conjunto de regiones potencialmente defectuosas, o nodos, $\mathcal{E} = \{e_1, e_2, \dots, e_n\}$ y una función de vecindad N , donde $N_i \subseteq \mathcal{E} \setminus \{e_i\}$, que describe la estructura de la red subyacente [NSBG09]. Siendo $\mathcal{E}$ una colección aleatoria de regiones potencialmente defectuosas, se divide en dos conjuntos $\mathcal{X}$ e $\mathcal{Y}$, donde $\mathcal{X}$ está formado por las regiones de las que conocemos sus valores correctos, e $\mathcal{Y}$ son las regiones para las que buscamos conocer su valor. Teniendo en cuenta todo esto, la tarea consiste en etiquetar los nodos $\mathcal{Y}_i \in \mathcal{Y}$ con la menor cantidad de etiquetas posibles $\mathcal{L} = \{l_1, l_2, \dots, l_q\}$.

En esta tesis, hemos utilizado los siguientes algoritmos pertenecientes a los clasificadores colectivos: (i) *Collective IBK* basado en el algoritmo KNN, (ii) *Collective Forest*, (iii) *Collective Woods*, y por último (iv) *RandomWoods*. Estos tres últimos, por su parte, se basan en árboles de decisión. Todos ellos están implementados como módulos para la herramienta WEKA [G⁺95].

5.2.1.1 *Collective IBK*

El clasificador *Collective IBK* utiliza la implementación del algoritmo IBK de WEKA, basado en KNN, para determinar el valor óptimo de K en el conjunto de entrenamiento. Posteriormente, este valor de K se usa para generar un ve-

cindario para cada una de las instancias del conjunto de prueba; basándose en instancias, tanto del conjunto de entrenamiento como del conjunto de prueba. Después, se ordenan los elementos de cada uno de los vecindarios en base a las distancias con respecto a las instancias de prueba a las que pertenecen. Finalmente, se determina el valor de la clase para cada una de las instancias, en base a la mayoría de los votos de los elementos del vecindario. En caso de que exista un empate en número de votos, se elige la primera de las clases.

Este proceso se realiza hasta que no queda ninguna instancia sin etiquetar en el conjunto de datos de prueba.

5.2.1.2 *Collective Forest*

Este clasificador utiliza la implementación del algoritmo *RandomTree* de WEKA como base para dividir el conjunto de datos de prueba en K subconjuntos distintos, estando formados todos ellos por el mismo número de elementos. En una primera iteración, se entrena el modelo usando, para ello, el conjunto de datos de entrenamiento original, y se genera la distribución para todas las instancias del conjunto de prueba. A continuación, se añaden las mejores instancias al conjunto de entrenamiento. Cabe destacar, que el número de instancias añadidas al modelo y cada subconjunto tiene que ser el mismo.

Las siguientes iteraciones entrenan el modelo con el nuevo conjunto de datos de entrenamiento y generan las distribuciones para las instancias restantes en el conjunto de datos de prueba.

5.2.1.3 *Collective Woods* y *Collective Tree*

El clasificador *Collective Woods* tiene un funcionamiento muy similar a *Collective Forest*, con la salvedad de que en vez de usar el algoritmo *RandomTree*, emplea el *Collective Tree*. En general, divide el conjunto de datos de prueba en distintos subconjuntos, y progresivamente los va añadiendo al conjunto de entrenamiento asignándoles la mejor predicción en cada caso.

Por su parte, *Collective Tree* funciona de forma muy similar a un árbol de decisión. El proceso comienza particionando el atributo en una posición tal, que divide el subconjunto actual, tanto de entrenamiento como de prueba, en dos mitades. El proceso finaliza cuando se cumple una de las siguientes condiciones: (i) se incluyen solo las instancias del conjunto de entrenamiento, cuyas etiquetas ya son conocidas; (ii) sólo hay instancias del conjunto de prueba en las hojas, todo ello, teniendo en cuenta la distribución del nodo padre; y por último, (iii) en caso de que todas las instancias del conjunto de prueba sean consideradas de una de las clases, el conjunto de entrenamiento estará compuesto por instancias de esa única clase.

Finalmente, para calcular la distribución final de la clase de un conjunto o subconjunto de datos, se suman los pesos, de acuerdo a los establecidos por

el conjunto de prueba, y se normalizan. De esta forma, la distribución de un atributo nominal se corresponde con la suma de los pesos normalizados para cada uno de los valores posibles, mientras que, la de un atributo numérico se calcula en base a la mediana de la división binaria de la distribución, a la que finalmente se le suman los pesos y se normalizan.

5.2.1.4 *RandomWoods*

Este clasificador funciona igual que *Random Forest*, pero sustituyendo *CollectiveBagging* por *Bagging* y *Collective Tree* en el lugar de *RandomTree*.

A lo largo de esta sección, hemos ido viendo cómo funcionan los distintos algoritmos de aprendizaje colectivo. Algo que comparten todos estos enfoques, es que sólo funcionan con problemas de clasificación binarios. En el ámbito de aplicación de esta tesis, esto supone una importante limitación, ya que nuestro problema de clasificación no es binario, sino que existen múltiples clases posibles. Por este motivo, a continuación, vamos a explorar los enfoques denominados multiclase.

5.2.2 Enfoques multiclase

Los enfoques multiclase de clasificación se basan en la descomposición de un problema de clasificación múltiple en varios, empleando para cada uno de ellos clasificadores binarios [A05]. Aunque, existe un gran número de enfoques en la literatura [DB95, ASS01, HL02], en esta tesis vamos a aplicar uno de los enfoques más sencillos y que mejores resultados ha demostrado en distintas áreas de aplicación [ASS01, HL02].

5.2.2.1 Todos contra todos

En este enfoque, también conocido como AVA (*All Versus All*), se compara cada una de las clases posibles contra todas las demás [HT98]. En este caso, se generan $(K(K - 1))/2$ clasificadores binarios, con lo que la evaluación de las clases se realiza por pares. A la hora de clasificar una nueva instancia desconocida, se realiza una votación entre todos los clasificadores eligiendo la etiqueta más votada.

5.2.3 Validación empírica de la mejora propuesta para el método de categorización

En esta sección vamos a analizar los resultados obtenidos al sustituir los algoritmos de aprendizaje supervisado por clasificadores de aprendizaje colectivo. Para ello, comenzaremos con la especificación de la metodología de experimentación y evaluación que vamos a seguir.

5.2.3.1 Metodología de evaluación y medición de los resultados

Para realizar la experimentación planteada, hemos vuelto a utilizar una vez más la herramienta WEKA [G⁺95], con la salvedad de que hemos empleado un nuevo módulo de clasificación colectiva¹. Este módulo incluye los algoritmos que hemos visto en la Sección 5.2.1. En la Tabla 5.6 mostramos los algoritmos empleados junto con las configuraciones.

Tabla 5.6: Algoritmos de aprendizaje colectivo empleados, junto con sus configuraciones.

Algoritmo	Configuración empleada
<i>Collective IBK</i>	$K = 5$
<i>Collective Forest</i>	$N = 100$
<i>Collective Woods</i>	$N = 100$
<i>Random Woods</i>	$N = 100$

Además, para evaluar los resultados de los algoritmos de aprendizaje colectivo, hemos entrenado cada uno de ellos con distintos porcentajes de instancias etiquetadas. Estos porcentajes son: 10 %, 20 %, 33 %, 50 %, 66 %, 75 %, 80 % y 90 %.

Así mismo, hemos probado a categorizar los defectos en base a los dos enfoques ya planteados en el Capítulo 4. De esta forma, evaluaremos cuál es la mejor estrategia de categorización a seguir: (i) directa o (ii) en dos fases. En ambos casos, hemos utilizado el enfoque multiclase de todos contra todos (AVA) en los experimentos con más de dos clases. En cuanto a la evaluación de los resultados obtenidos, hemos utilizado las medidas ya descritas en la Sección 4.3.2, entre las que se encuentran: el TPR, el FPR, el ATPR, el AFPR, la precisión, y el área bajo la curva ROC.

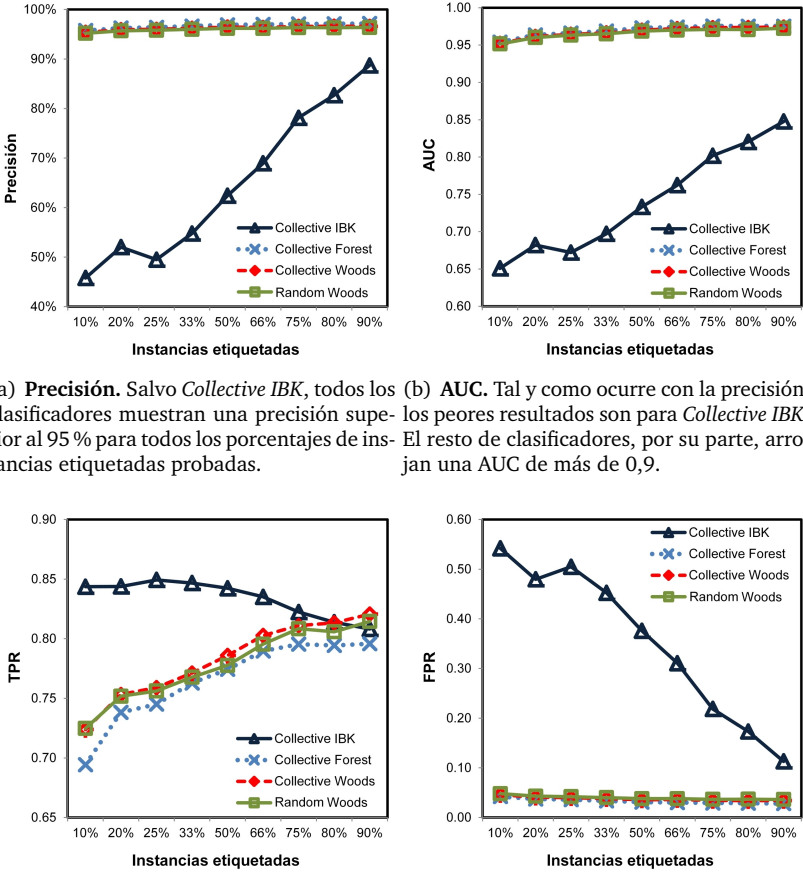
Por otra parte, hemos realizado de nuevo un estudio para evaluar cuál es la mejor combinación de características, pero en este caso, hemos eliminado los conjuntos que han obtenido los resultados más bajos con los algoritmos de aprendizaje supervisado. Entre ellos se encuentran los siguientes grupos de características: (i) las simples, (ii) las simples con las extraídas del *Best Crossing Line Profile*, (iii) las simples con las extraídas de la transformada rápida de Fourier, (iv) las simples con las extraídas de la matriz de co-ocurrencia y (v) todas las características juntas.

Por último, y debido a que el conjunto de datos se encuentra desbalanceado, hemos vuelto a aplicar la técnica SMOTE sobre el conjunto de entrenamiento (véase la Sección 4.3.1).

¹Herramienta disponible en: <http://www.cms.waikato.ac.nz/~fracpete/projects/collective-classification>

5.2.3.2 Resultados obtenidos mediante la experimentación

- Categorización en dos fases:
 - Fase 1 - Características simples.

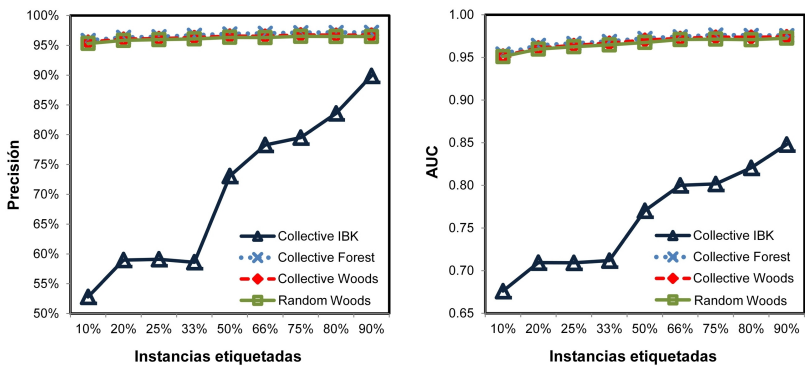


(a) **Precisión.** Salvo *Collective IBK*, todos los clasificadores muestran una precisión superior al 95 % para todos los porcentajes de instancias etiquetadas probadas. (b) **AUC.** Tal y como ocurre con la precisión, los peores resultados son para *Collective IBK*. El resto de clasificadores, por su parte, arrojan una AUC de más de 0,9.

(c) **TPR.** En general, el TPR ha aumentado a mayor porcentaje de instancias etiquetadas. El mejor clasificador ha sido *Collective Woods*, que con el 25 % de instancias etiquetadas, muestra un TPR de más de 0,75. (d) **FPR.** *Collective IBK* ha sido el peor clasificador, ya que no ha bajado de 0,1 ni con el 90 % de las instancias etiquetadas. Por su parte, el resto de clasificadores obtienen un FPR de menos de 0,05.

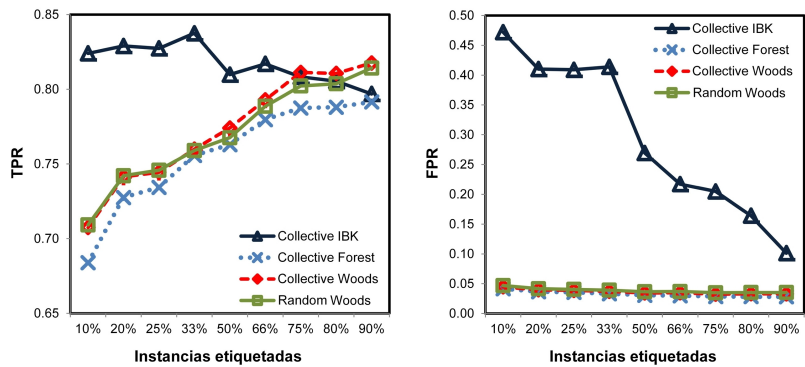
Figura 5.4: Resultados obtenidos en términos de precisión (en porcentaje), TPR, FPR y AUC para los distintos porcentajes de instancias etiquetadas.

- Fase 1 - Características simples combinadas con las extraídas del *Best Crossing Line Profile*.



(a) **Precisión.** Todos los clasificadores arrojan precisiones muy similares que rondan el 95%, exceptuando *Collective IBK*, que no llega al 90% para ninguno de los porcentajes probados.

(b) **AUC.** *Collective IBK* muestra los peores resultados para la AUC. El resto, por su parte, superan el 0,95 para porcentajes de instancias etiquetadas iguales o superiores al 20%.

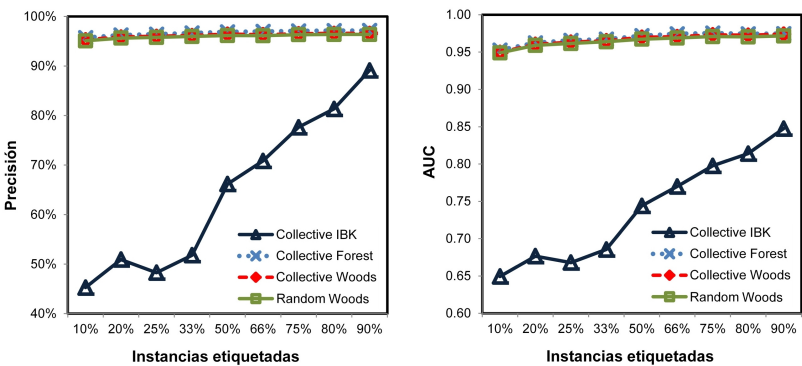


(c) **TPR.** Los clasificadores se han mostrado sensibles al porcentaje de instancias etiquetadas. Los mejores resultados se obtienen con *Collective Woods*, y más concretamente, que sólo logra bajar del 0,1 con el 90% de las instancias etiquetadas.

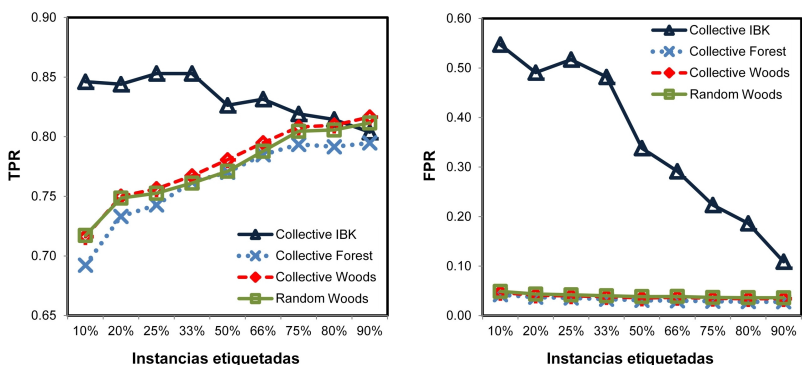
(d) **FPR.** En general todos los clasificadores arrojan unos buenos resultados con tasas inferiores a 0,05, exceptuando *Collective IBK*, que sólo logra bajar del 0,1 con el 90% de las instancias etiquetadas.

Figura 5.5: Resultados obtenidos en términos de precisión (en porcentaje), TPR, FPR y AUC para los distintos porcentajes de instancias etiquetadas.

- Fase 1 - Características simples combinadas con las extraídas de la transformada rápida de Fourier.



(a) **Precisión.** Las precisiones, en general, (b) **AUC.** De forma similar a lo que ocurre con los resultados de la precisión, todos los clasificadores empleados, exceptuando los algoritmos a excepción de *Collective IBK*, que no llega al 90 % en ningún caso.

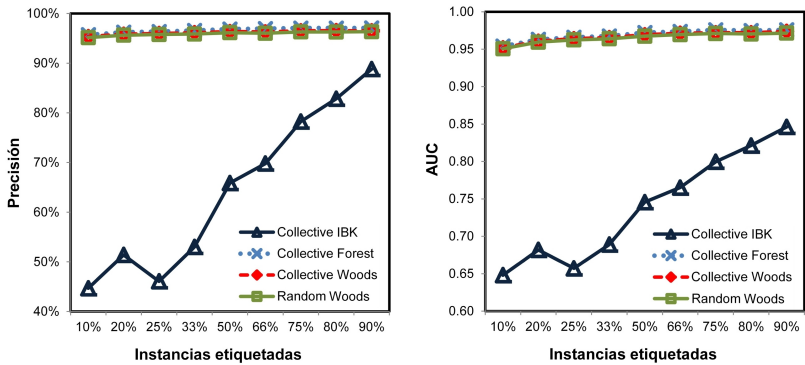


(c) **TPR.** El clasificador con mejores resultados es *Collective Woods*, que con un 75 % de instancias etiquetadas supera el 0,8. Otro de los clasificadores destacados es *Random Woods* que con el mismo porcentaje, ronda un TPR de 0,8.

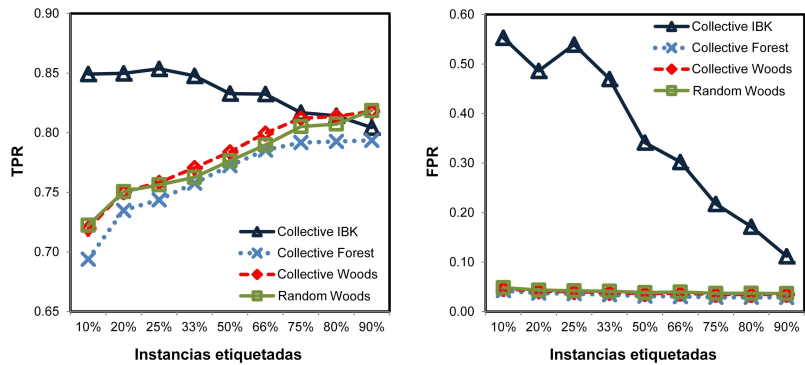
(d) **FPR.** En este caso, todos los clasificadores muestran un FPR inferior a 0,05, a excepción de *Collective IBK*, que con el 90 % de instancias etiquetadas no baja de 0,1.

Figura 5.6: Resultados obtenidos en términos de precisión (en porcentaje), TPR, FPR y AUC para los distintos porcentajes de instancias etiquetadas.

- Fase 1 - Características simples combinadas con las extraídas de la matriz de co-ocurrencia.



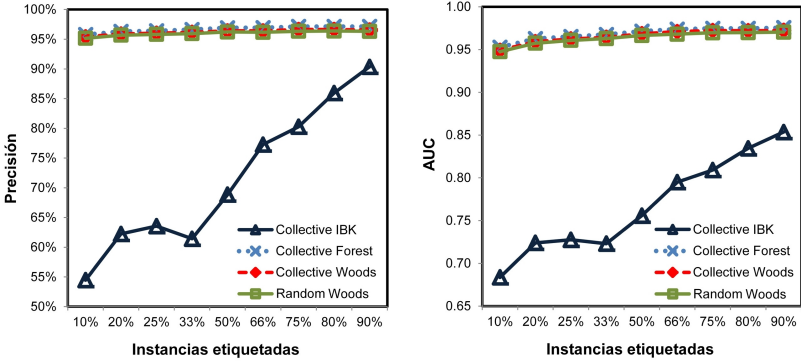
(a) **Precisión.** En general, la precisión supera el 95%. El clasificador que arroja peores resultados es *Collective IBK*, que no supera el 90% de precisión en ningún caso. (b) **AUC.** Tal y como ocurre con la precisión, los valores para AUC están por encima del 0,95 a excepción de *Collective IBK*, que arroja valores por debajo de 0,85.



(c) **TPR.** Los resultados varían en base al porcentaje de instancias etiquetadas, esto es, los arroja *Collective IBK*, que no baja de a mayor porcentaje, mayor TPR. *Collective Woods* supera el 0,8 de TPR con tan sólo un 66% de las instancias etiquetadas, convirtiéndose, en el mejor de los clasificadores. (d) **FPR.** El clasificador que peor resultado arroja es *Collective IBK*, que no baja de un FPR de 0,10. El resto, por su parte, han obtenido resultados similares, sin superar en ningún caso un FPR del 0,05.

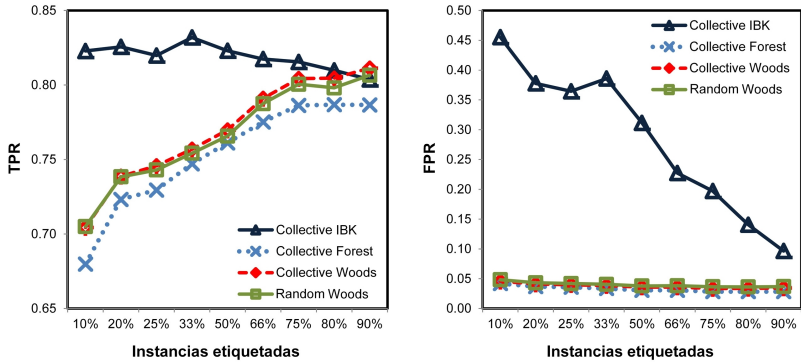
Figura 5.7: Resultados obtenidos en términos de precisión (en porcentaje), TPR, FPR y AUC para los distintos porcentajes de instancias etiquetadas.

– Fase 1 - Todas las características



(a) **Precisión.** La precisión, por regla general, se mantiene por encima del 95 % para todos los clasificadores, salvo para *Collective IBK*, que no supera el 90 % para ninguno de los porcentajes probados.

(b) **AUC.** De la misma forma que ocurre para la precisión, a excepción de *Collective IBK*, todos los demás muestran valores de AUC por encima de 0,95.

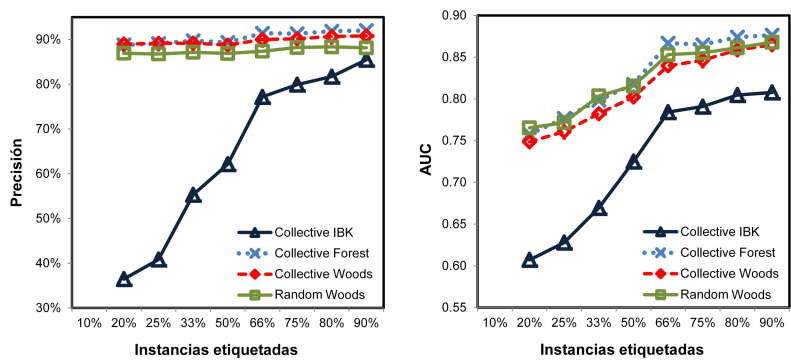


(c) **TPR.** Tanto *Collective Woods* como *Random Woods* muestran unos valores para TPR por debajo de 0,05, excepto *Collective IBK*, que muestra valores superiores a 0,8.

(d) **FPR.** Los valores se mantienen por debajo de 0,05, excepto *Collective IBK*, que muestra valores superiores a 0,4.

Figura 5.8: Resultados obtenidos en términos de precisión (en porcentaje), TPR, FPR y AUC para los distintos porcentajes de instancias etiquetadas.

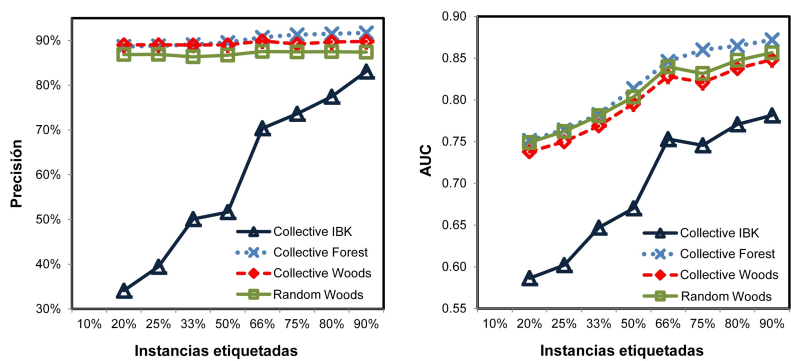
– Fase 2 - Características simples.



(a) **Precisión.** El clasificador que mejores resultados muestra es *Collective Forest*, que con un 66% etiquetado supera el 90% de precisión. (b) **AUC.** Del mismo modo que ocurre con la precisión, el clasificador *Collective Forest* con un 66% de las instancias etiquetadas obtiene una AUC de 0,87.

Figura 5.9: Resultados obtenidos en términos de precisión (en porcentaje) y AUC para los distintos porcentajes de instancias etiquetadas.

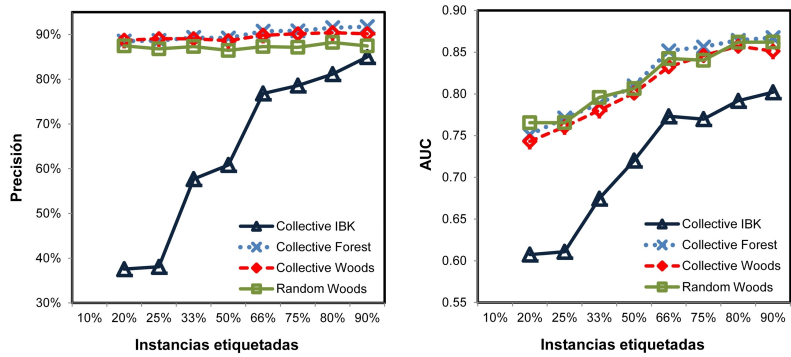
– Fase 2 - Características simples combinadas con las extraídas del *Best Crossing Line Profile*.



(a) **Precisión.** Todos los clasificadores, excepto *Collective IBK*, muestran unas precisiones superiores al 85%. (b) **AUC.** El mejor de los clasificadores es el *Collective Forest*, arrojando una AUC de 0,86 con el 75% de las instancias etiquetadas.

Figura 5.10: Resultados obtenidos en términos de precisión (en porcentaje) y AUC para los distintos porcentajes de instancias etiquetadas.

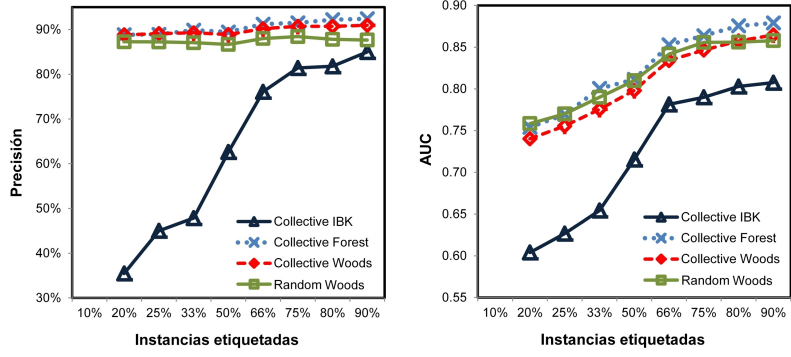
- Fase 2 - Características simples combinadas con las extraídas de la transformada rápida de Fourier.



(a) **Precisión.** Destacan los *Collective Forest* y *Woods*, que superan el 90 % de precisión con el 66 % etiquetado. (b) **AUC.** El mejor resultado es para *Collective Forest* con un valor de 0,85 para un 66 % de las instancias etiquetadas.

Figura 5.11: Resultados obtenidos en términos de precisión (en porcentaje) y AUC para los distintos porcentajes de instancias etiquetadas.

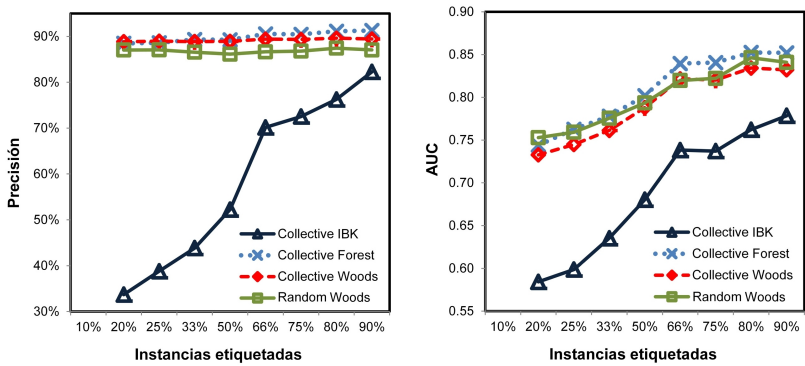
- Fase 2 - Características simples combinadas con las extraídas de la matriz de co-ocurrencia.



(a) **Precisión.** El mejor es *Collective Forest*, que con un porcentaje de datos etiquetados del 33 %, supera el 90 % de precisión. (b) **AUC.** Destaca *Collective Forest* con una AUC de más de 0,85 con el 66 % etiquetado.

Figura 5.12: Resultados obtenidos en términos de precisión (en porcentaje) y AUC para los distintos porcentajes de instancias etiquetadas.

– Fase 2 - Todas las características.



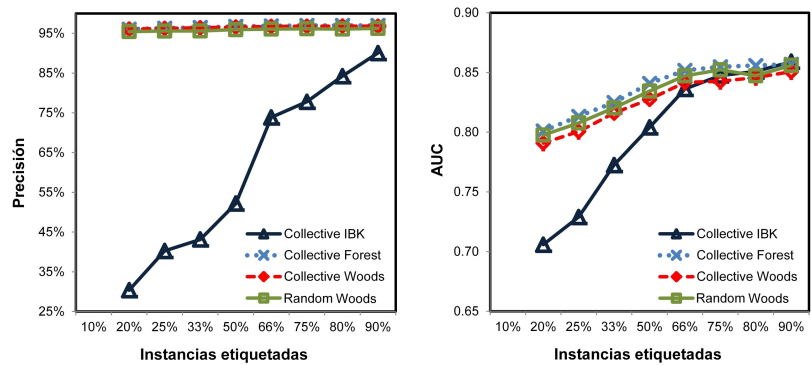
(a) **Precisión.** *Collective Forest* muestra de (b) **AUC.** Los mejores valores para la AUC los mejores resultados, superando el 90% de son para *Collective Forest*, que con el 66% precisión para el 66% de las instancias eti- etiquetado se aproxima al 0,85. quetadas.

Figura 5.13: Resultados obtenidos en términos de precisión (en porcentaje) y AUC para los distintos porcentajes de instancias etiquetadas.

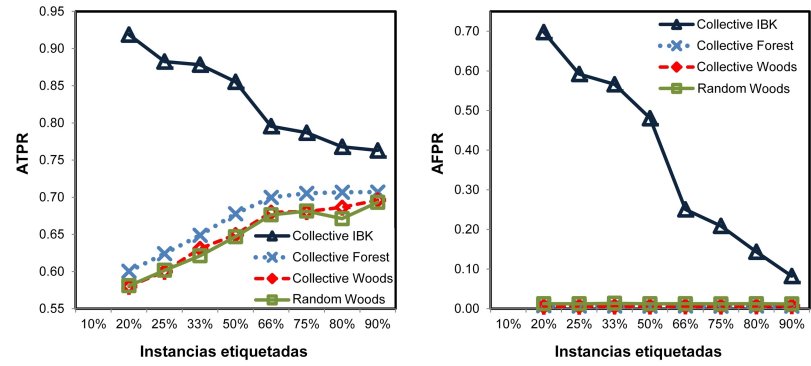
Tal y como hemos podido observar en los resultados obtenidos hasta el momento, los clasificadores se han mostrado sensibles a la cantidad de instancias de entrenamiento etiquetadas; esto es, a más datos etiquetados obtenemos unos resultados mejores. Un patrón común, que se ha repetido en todos los resultados, es que los clasificadores basados en árboles de decisión se han comportado mejor que el algoritmo *Collective IBK*.

Por último, queremos destacar que hemos escalado los ejes de las gráficas en cada caso para poder apreciar mejor las diferencias entre los resultados de cada uno de los clasificadores.

- Categorización directa:
 - Características simples.



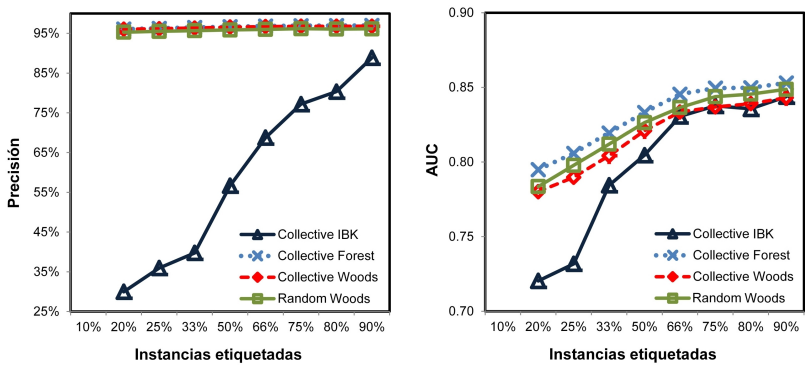
(a) **Precisión.** Tanto *Collective Forest*, como (b) **AUC.** En este caso, los mejores resultados son para *Collective Forest* y para *Random Woods*, que con un porcentaje de etiquetado de los porcentajes probados. *Collective IBK*, del 66% consiguen una AUC de 0,85. por su parte, no supera en ningún caso el 90%.



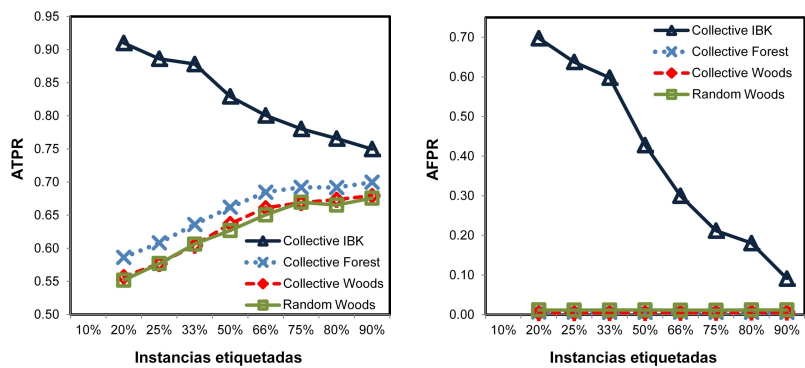
(c) **ATPR.** El mejor clasificador ha sido *Collective Forest*, aproximándose al valor de un AFPR inferior a 0,05, excepto *Collective IBK*, que arroja los peores resultados. (d) **AFPR.** Todos los clasificadores muestran un AFPR inferior a 0,05, excepto *Collective IBK*, que arroja los peores resultados.

Figura 5.14: Resultados obtenidos en términos de precisión (en porcentaje), ATPR, AFPR y AUC para los distintos porcentajes de instancias etiquetadas.

- Características simples combinadas con las características del *Best Crossing Line Profile*.



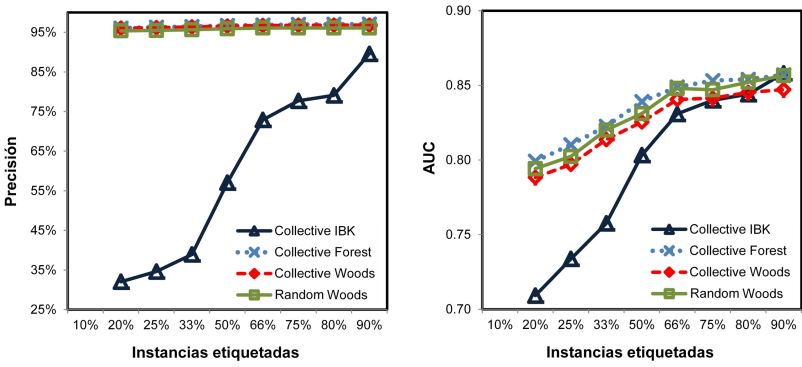
(a) **Precisión.** *Collective IBK* es el clasifica- (b) **AUC.** El clasificador que arroja los me-
dor que más sensibilidad muestra al cambiar jores resultados es *Collective Forest*, aunque
los porcentajes de instancias etiquetadas. El también son destacables los valores obteni-
resto, por su parte, se mantienen por encima dos por *Random Woods*.
del 95% de precisión.



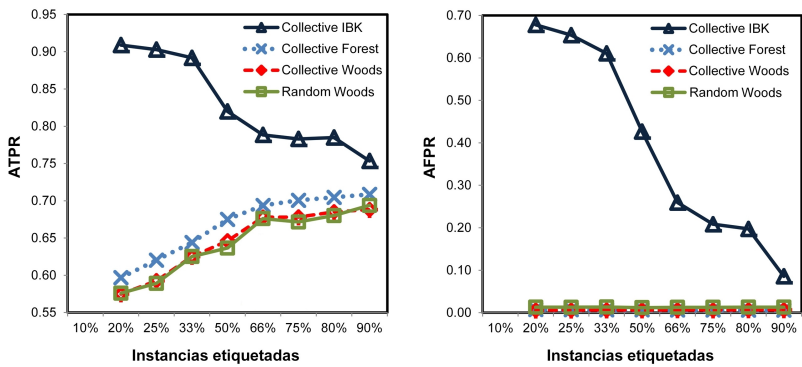
(c) **ATPR.** En general, el ATPR ha aumen- (d) **AFPR.** *Collective IBK* ha sido el peor cla-
tado a más instancias etiquetadas. El mejor sificador, ya que únicamente ha bajado de
clasificador ha sido *Collective Forest*, que con 0,1 con el 90% de las instancias etiqueta-
el 75% de instancias etiquetadas, muestra das. Por su parte, el resto de clasificadores
un ATPR de más de 0,70. obtienen un AFPR de menos de 0,03.

Figura 5.15: Resultados obtenidos en términos de precisión (en porcentaje), ATPR, AFPR y AUC para los distintos porcentajes de instancias etiquetadas.

- Características simples combinadas con las características de la transformada rápida de Fourier.



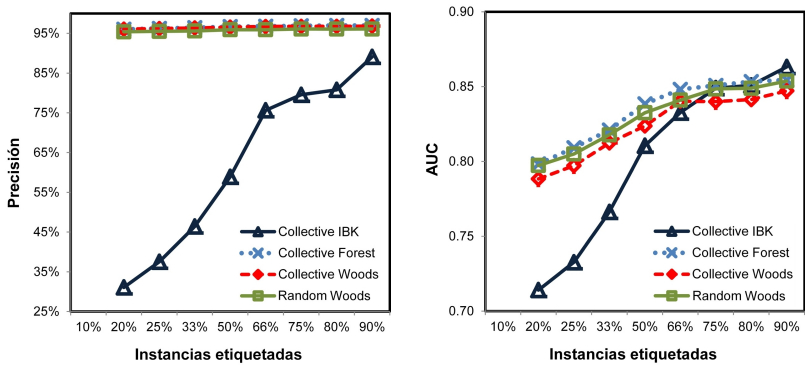
(a) **Precisión.** Todos los clasificadores arro- (b) **AUC.** *Collective IBK* muestra los peores
jan precisiones muy similares que rondan el valores para la AUC. El resto, por su parte,
95%, exceptuando *Collective IBK*, que sólo superan el 0,80 para porcentajes de instan-
alcanza el 90% de precisión con el 90% de cias etiquetadas iguales o superiores al 33%
instancias etiquetadas.



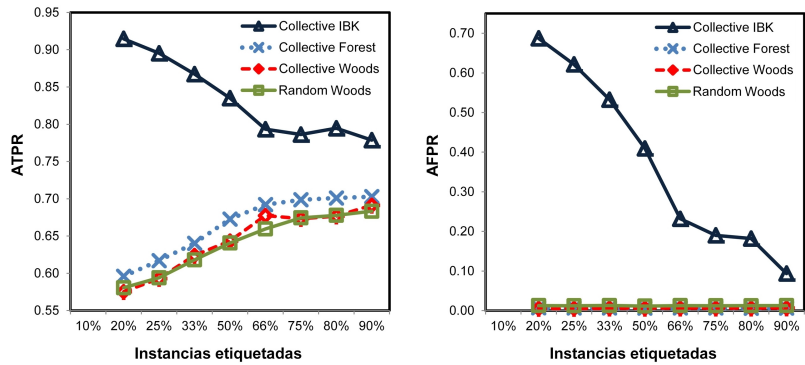
(c) **ATPR.** Los clasificadores se han mostra- (d) **AFPR.** En general todos los clasificado-
do sensibles al porcentaje de instancias eti- res arrojan unos buenos resultados con tasas
quetadas. Los mejores resultados se obtienen inferiores a 0,03, exceptuando *Collective IBK*,
con *Collective Forest*, y más concretamente, que sólo logra bajar del 0,1 con el 90% de las
con un porcentaje de etiquetado superior al instancias etiquetadas.
66%.

Figura 5.16: Resultados obtenidos en términos de precisión (en porcentaje), ATPR, AFPR y AUC para los distintos porcentajes de instancias etiquetadas.

- Características simples combinadas con las características de la matriz de co-ocurrencia.



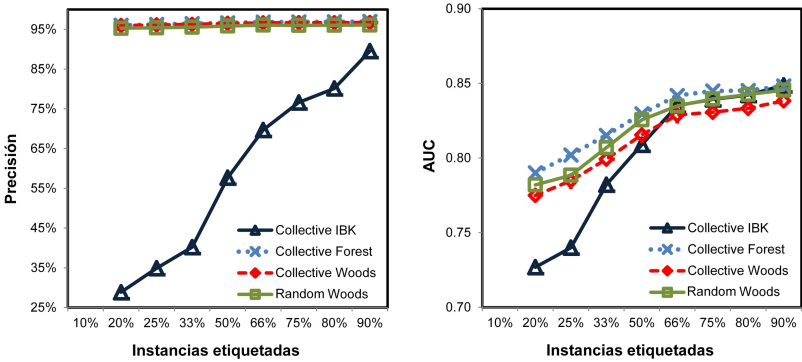
(a) **Precisión.** Las precisiones, en general, se mantienen por encima del 95 % para todos los clasificadores empleados, exceptuando *Collective IBK*, que no llega al 90 % en ningún caso. (b) **AUC.** En este caso, todos los clasificados se aproximan a un valor de 0,85 cuando el porcentaje de instancias supera el 66 %



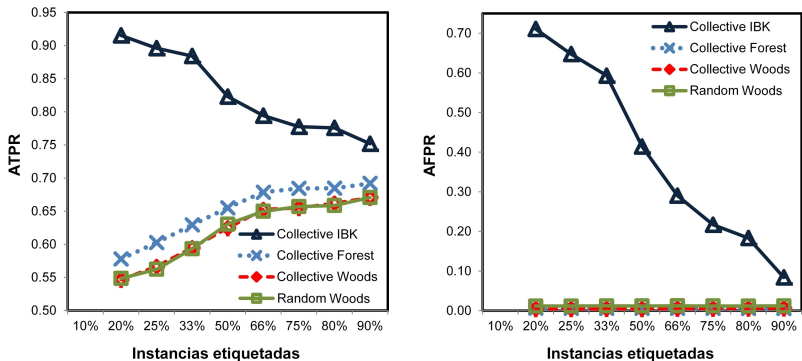
(c) **ATPR.** El clasificador con mejores resultados es *Collective Forest*, que con un 75 % de instancias etiquetadas se aproxima a un ATPR del 0,70. (d) **FPR.** En este caso, todos los clasificados muestran un AFPR inferior a 0,02, a excepción de *Collective IBK*, que con el 90 % de instancias etiquetadas alcanza el 0,1.

Figura 5.17: Resultados obtenidos en términos de precisión (en porcentaje), ATPR, AFPR y AUC para los distintos porcentajes de instancias etiquetadas.

– Todas las características.



(a) **Precisión.** En general, la precisión supera el 95%. El clasificador que arroja peores resultados es *Collective IBK*, que no supera el 90% de precisión en ningún caso. (b) **AUC.** Tanto *Collective Forest*, como *Collective Woods* y *Collective IBK* mantienen sus valores para la AUC por encima de 0,8 para todos los porcentajes de instancias etiquetadas superiores al 50%.



(c) **ATPR.** Los resultados varían en base al porcentaje de instancias etiquetadas, esto es, dos arroja es *Collective IBK*. El resto, por su a mayor porcentaje, el ATPR es más alto. *Collective Forest* supera el 0,65 de ATPR con tan sólo un 66% de las instancias etiquetadas, convirtiéndose, en el mejor de los clasificadores. (d) **FPR.** El clasificador que peores resultados arroja es *Collective IBK*. El resto, por su parte, han obtenido resultados similares, sin superar en ningún caso un AFPR del 0,03.

Figura 5.18: Resultados obtenidos en términos de precisión (en porcentaje), TPR, FPR y AUC para los distintos porcentajes de instancias etiquetadas.

Como ha ocurrido para la categorización en dos fases, los clasificadores basados en árboles de decisión han mostrado mejores resultados que el algoritmo *Collective IBK*. Además, hemos vuelto a escalar los ejes de las gráficas

para poder comparar de un forma mejor la evolución de los distintos clasificadores con los porcentajes de instancias etiquetadas probados.

5.3 Discusión de los resultados

Los resultados obtenidos mediante las configuraciones planteadas han demostrado que es posible detectar y categorizar defectos superficiales en piezas fundición de hierro nodular, empleando para ello clasificadores basados en aprendizaje colectivo. A continuación, vamos a evaluar los resultados obtenidos para las dos estrategias de resolución y para los distintos conjuntos de características extraídas. Además, valoraremos la mejora lograda en base a la configuración óptima del Capítulo 4.

Tabla 5.7: Comparación de los mejores clasificadores supervisados y colectivos en términos de precisión, TPR, FPR, y AUC para los mejores conjuntos de características empleadas, en cada caso, para la fase 1 de la categorización en dos fases.

Algoritmo	Características empleadas	Prec.(%)	TPR	FPR	AUC
SVM: Pearson VII	Simple + matriz de co-ocurrencia	97,29	0,8196	0,0271	0,9770
AD: Random Forest N=100	Simple + matriz de co-ocurrencia	97,02	0,7922	0,0298	0,9749
Collective Forest (66 %)	Simple	96,99	0,7897	0,0301	0,9740
Collective Forest (75 %)	Simple	97,08	0,7955	0,0292	0,9756
Collective Forest (80 %)	Simple	97,13	0,7944	0,0287	0,9754

Como podemos observar en la Tabla 5.7, los resultados obtenidos por *Collective Woods* con un porcentaje del 66 % de instancias etiquetadas, se aproximan bastante a los arrojados por *Random Forest*. Si incrementamos este porcentaje hasta 75 %, vemos que los resultados son superiores en el caso del clasificador colectivo. Por el contrario, si la comparación la realizamos contra el clasificador SVM con kernel Pearson VII, los resultados son sensiblemente más bajos, aunque reducimos el esfuerzo de etiquetado en 25 puntos porcentuales.

En cuanto a los resultados de la segunda fase, las precisiones se han mantenido muy similares entre si. En cambio, si observamos los valores para el área bajo la curva ROC, los algoritmos de aprendizaje supervisado muestran valores muy superiores a los obtenidos por los clasificadores colectivos.

En la Tabla 5.9 vemos los resultados para la categorización directa. En este caso, *Collective Woods* consigue una precisión del 95,74, un ATPR de 0,8071, un AFPR de 0,0246 y un área bajo la curva ROC de 0,8954, con tan sólo el 66 % de las instancias etiquetadas. Concretamente, la precisión, el ATPR y el

Tabla 5.8: Comparación de los mejores clasificadores supervisados y colectivos en términos de precisión y AUC para los mejores conjuntos de características empleadas, en cada caso, para la fase 2 de la categorización en dos fases.

Algoritmo	Características empleadas	Prec.(%)	AUC
<i>SVM:</i> <i>Pearson VII</i>	Simple + matriz de co-ocurrencia	88,22	0,9070
<i>AD: Random</i> <i>Forest N=100</i>	Simple + matriz de co-ocurrencia	91,96	0,9633
<i>Collective</i> <i>Forest (66 %)</i>	Simple	91,40	0,8668
<i>Collective</i> <i>Forest (75 %)</i>	Simple	91,29	0,8649
<i>Collective</i> <i>Forest (80 %)</i>	Simple	91,85	0,8740

AFPR se han mantenido, mientras que como ocurre en la segunda fase de la categorización en 2 fases, la AUC ha descendido notablemente.

Tabla 5.9: Comparación de los mejores clasificadores supervisados y colectivos en términos de precisión, ATPR, AFPR, y AUC para los mejores conjuntos de características empleadas, en cada caso, para la categorización directa.

Algoritmo	Características empleadas	Prec.(%)	ATPR	AFPR	AUC
<i>SVM:</i> <i>Pearson VII</i>	Simple + transformada Fourier	96,42	0,8250	0,0252	0,9774
<i>AD: Random</i> <i>Forest N=100</i>	Simple + transformada Fourier	96,61	0,7844	0,0197	0,9760
<i>Collective</i> <i>Woods (66 %)</i>	Simple + transformada Fourier	95,74	0,8071	0,0246	0,8954
<i>Collective</i> <i>Woods (75 %)</i>	Simple + transformada Fourier	95,95	0,8052	0,0224	0,8977
<i>Collective</i> <i>Woods (80 %)</i>	Simple + transformada Fourier	95,92	0,8131	0,0236	0,9011

Como hemos podido observar, los algoritmos colectivos han mantenido generalmente los resultados presentados por los clasificadores supervisados. Sin embargo, hemos detectado un descenso considerable en el valor del área bajo la curva ROC para las clasificaciones compuestas por más de dos clases. Esto, se debe principalmente a la estrategia de multclasificación que hemos seguido, todos contra todos.

A la vista de los resultados obtenidos, podemos afirmar que los clasificadores colectivos pueden sustituir a los algoritmos supervisados disminuyendo, de esta forma, el esfuerzo de etiquetado. En concreto, hemos conseguido disminuir en 33 puntos porcentuales la cantidad de datos etiquetados para la fase de aprendizaje, manteniendo prácticamente la misma precisión.

5.4 Sumario

A lo largo de este capítulo, hemos propuesto dos mejoras para el sistema planteado en esta tesis doctoral. La primera de ellas es la relativa al método de segmentación expuesto en el Capítulo 3. Concretamente, hemos aplicado técnicas de *clustering* para reducir el número de piezas del modelo que emplea el sistema. Además, con esta mejora hemos conseguido incrementar el porcentaje de defectos detectados, ya que las piezas centroides generadas poseen menos ruido que una pieza normal.

La segunda mejora que hemos propuesto está asociada al método de categorización de regiones planteada en el Capítulo 4. Para ello, hemos cambiado los algoritmos de aprendizaje automático supervisado por algoritmos de aprendizaje colectivo. Con este cambio hemos reducido la cantidad de instancias etiquetadas que requiere el método de categorización para funcionar correctamente.

Finalmente, destacamos que los resultados arrojados por la experimentación realizada, demuestran la validez de las mejoras propuestas. En el caso del método de segmentación, hemos mejorado los resultados obtenidos, y en el caso del método de categorización, hemos mantenido los resultados de los que partíamos reduciendo, a su vez, la labor de etiquetar los datos para la fase de aprendizaje.

«A veces sentimos que lo que hacemos es tan solo una gota en el mar, pero el mar sería menos si le faltara una gota.»

Madre Teresa de Calcuta (1910–1997)

CAPÍTULO

6

Conclusiones

A lo largo de esta investigación, hemos ido identificando las distintas problemáticas que pertenecen al ámbito de la inspección de los defectos superficiales en las fundiciones de hierro nodular. En este contexto, repasamos las contribuciones realizadas a través de esta tesis doctoral para evaluar el grado de cumplimiento de cada uno de los objetivos.

El resto del capítulo continúa de la siguiente forma. La Sección 6.1 describe de forma detallada las contribuciones principales que se han realizado a lo largo de esta tesis doctoral, así como la evaluación del cumplimiento de los objetivos fijados. La Sección 6.2 expone las limitaciones que hemos encontrado durante el proceso de investigación llevado a cabo. La Sección 6.3 plantea las líneas futuras que surgen a raíz de la investigación realizada. Por último, la Sección 6.4 recoge las consideraciones finales de esta tesis doctoral.

6.1 Resumen de los resultados

A continuación vamos a enumerar las contribuciones principales que hemos realizado mediante la presente tesis doctoral. En concreto, son las siguientes: (i) el diseño e implementación de un sistema de visión artificial para la captura de la información superficial de piezas de fundición; (ii) un nuevo enfoque para la representación de la información de piezas de fundición; (iii) un nuevo método de segmentación para la detección de posibles zonas de interés en la superficie; (iv) una nueva representación de las características más relevantes

de las regiones potencialmente defectuosas de la superficie de las piezas de fundición; (v) un nuevo método de categorización de regiones basado en dos estrategias, una directa y otra en dos fases; (vi) una mejora del método de segmentación basado en técnicas de *clustering*; y por último, (vii) una mejora para el método de categorización basado en técnicas de aprendizaje colectivo.

1. **Un sistema de visión artificial para la captura de los datos superficiales de las piezas de fundición.** El sistema de visión artificial que hemos diseñado e implementado está compuesto por un dispositivo de captura tridimensional, un brazo robótico y una estación de trabajo de alta capacidad de cómputo. Mediante este sistema, capturamos los datos tomográficos de la superficie de las piezas de fundición. Posteriormente, tratamos estos datos para eliminar el ruido que se haya podido producir en la captura, así como las zonas que no pertenezcan a la superficie de la pieza. Finalmente, normalizamos la información para facilitar los procesos posteriores.
2. **Un enfoque para la representación de la información de las piezas de fundición.** Una vez que tenemos la información superficial normalizada, comenzamos a generar las representaciones que contienen toda la información relevante de las piezas. Estas representaciones son: (i) la matriz de alturas, con información dimensional en cada punto de la superficie; (ii) el mapa de alturas en escala de grises, que representa de forma gráfica los valores de la matriz de alturas en una escala compuesta por 256 tonos de gris; (iii) la matriz de vectores normales, que en cada punto representa los valores para las componentes (x , y , z) del vector normal a la superficie en cada punto de la matriz de alturas; (iv) y por último, el mapa de vectores normales codificado en RGB, donde representamos la matriz de vectores normales mediante una imagen, correspondiendo la x a la R, la y a la G, y la z a la B, obteniendo de esta forma una representación gráfica del mapa de vectores normales.
3. **Un método de segmentación para la detección de posibles zonas de interés en la superficie.** Hemos creado un método de segmentación basado en filtros de imagen que, empleando las representaciones generadas y modelos previamente almacenados, detecta zonas con variaciones fuera de lo común. De esta forma, eliminamos información no relevante de las superficies, y centramos los esfuerzos en zonas con una mayor probabilidad de ser defectuosas.
4. **Una representación de las características más relevantes de las regiones potencialmente defectuosas de la superficie de las piezas de fundición.** Una vez delimitadas las regiones potencialmente defectuosas, extraemos distintos grupos de características. Entre ellas encontramos: (i) las simples, que son aquellas que no requieren de un procesamiento extra; (ii) las obtenidas a partir del *Best Crossing Line Profile*, que buscan

diferenciar estructuras regulares de posibles defectos; (iii) las obtenidas a partir de la transformada rápida de Fourier, que se obtienen a partir de una representación de la región en el dominio frecuencial; y (iv) las obtenidas de la matriz de co-ocurrencia, que busca caracterizar una textura mediante los elementos de una matriz.

5. **Un método de categorización de regiones basado en dos estrategias, una directa y otra en dos fases.** Este método se encarga de determinar el tipo de defecto que pueda existir dentro de una región. Para ello, aplica algoritmos de aprendizaje automático supervisado que categorizan las regiones en base a las características extraídas de las mismas. En este punto hemos seguido dos estrategias de resolución distintas. La primera de ellas se basa en la división de un problema de clasificación multiclase en una binaria, determinando si la región es tolerable o defectuosa, y posteriormente, en caso de que sea defectuosa, indicando el tipo de defecto existente. La segunda de las estrategias determina de forma directa si la región es tolerable, o si contiene alguno de los defectos estudiados.
6. **Una mejora para el método de segmentación basado en técnicas de *clustering*.** Ésta, busca reducir el número de modelos que requiere el método, aumentando de esta forma el rendimiento del mismo. Además, hemos conseguido reducir información no relevante de las superficies modelo, aumentando a su vez, la precisión del método. Posteriormente, hemos evaluado el comportamiento del método de categorización anteriormente propuesto con las nuevas regiones generadas.
7. **Una mejora para el método de categorización basado en técnicas de aprendizaje colectivo.** Otra de las debilidades encontradas, es la asociada con los algoritmos de aprendizaje automático supervisado que se emplean, ya que requieren de un gran volumen de datos etiquetados para la fase de aprendizaje. En concreto, hemos propuesto un nuevo método que se basa en el uso de algoritmos de aprendizaje colectivo, minimizando de esta forma el esfuerzo de etiquetado.

Gracias a estas contribuciones, consideramos cumplidos todos los objetivos operacionales que planteamos en la Sección 1.4, y que recordamos a continuación:

- Diseñar, implementar y evaluar un sistema de visión artificial completo, que sea capaz de capturar la información superficial de las piezas a examinar.
- Desarrollar un método para generar representaciones normalizadas de la información.

- Desarrollar y evaluar un método de segmentación de imágenes, que indique las regiones de la superficie que puedan ser susceptibles de contener un defecto.
- Extraer y evaluar las características más representativas de las regiones segmentadas para determinar su importancia en la categorización.
- Desarrollar y evaluar un categorizador de regiones para determinar la validez de la pieza.

Así mismo, hemos logrado alcanzar los objetivos específicos planteados en la Sección 1.4, que son:

- Construir un sistema de visión artificial para la captura de los datos relativos a la superficie de las piezas de fundición.
- Desarrollar y evaluar un detector de zonas potencialmente defectuosas basado en técnicas de segmentación de imágenes.
- Desarrollar y evaluar un categorizador de defectos superficiales basado en técnicas de aprendizaje automático.

Con la consecución de todos estos objetivos damos por cumplido el objetivo principal de la presente tesis doctoral:

- Desarrollar y evaluar un detector y categorizador de defectos superficiales para piezas de fundición de hierro nodular.

Y como consecuencia de todo ello, damos por validada la hipótesis fundamental planteada en el Capítulo 1:

«Es posible, empleando técnicas de segmentación en imágenes y herramientas de aprendizaje automático, determinar la validez de una pieza de fundición de hierro nodular, detectando e identificando diferentes tipos de defectos superficiales.»

6.2 Limitaciones

La limitación principal que hemos encontrado en este proceso de investigación, es la incapacidad de automatizar por completo el sistema de inspección de piezas. Esto se debe principalmente a la complejidad del propio proceso de inspección en sí, ya que se requiere un sistema que sea capaz de coger las piezas independientemente de su geometría, tamaño y peso. En el Capítulo 2, hemos presentado diversos enfoques para la automatización de sistemas de visión artificial; todos ellos buscan automatizar el funcionamiento completo

del sistema, por lo que emplean diversos automatismos como cintas transportadoras o brazos robóticos.

En nuestro ámbito concreto de aplicación, no hemos encontrado ninguna solución que solvete esta problemática, ya que el sistema debe de ser capaz de manipular las piezas, y colocarlas en diversas posiciones para realizar las capturas necesarias, cubriendo de esta forma la mayor cantidad de superficie posible. Además, una vez analizada y determinada la validez de la pieza, este sistema de automatización debe ser capaz de separar aquellas que no cumplan los requisitos de calidad previamente establecidos.

6.3 Trabajo futuro

En esta sección vamos a desglosar las líneas de trabajo futuro que hemos identificado en torno a la presente tesis doctoral, siendo éstas: (i) mejorar el hardware del sistema de visión artificial propuesto, (ii) mejorar el método de segmentación de imágenes, y (iii) mejorar el método de categorización de defectos.

6.3.1 Mejorar el hardware del sistema de visión artificial propuesto

La primera línea de trabajo futuro se centra en la mejora del hardware que compone el sistema propuesto en el Capítulo 3. Para ello, habría que plantear estudios periódicos para determinar que tecnologías se adaptan mejor a los requisitos del sistema.

Atendiendo al dispositivo de captura, la tecnología basada en la triangulación láser es la que más precisión demuestra actualmente. A pesar de ello, los dispositivos basados en interferiometría están ganando peso en los últimos años en el sector industrial. Esto se debe principalmente, a su bajo coste en comparación con la tecnología láser y a su considerable aumento de precisión [BW10]. Además, son más rápidos que los dispositivos láser y no requieren tantas medidas de seguridad para funcionar [STD09].

En cuanto al dispositivo de automatización, tal y como hemos comentado en la Sección 6.2, el objetivo que se persigue desde el sector industrial es automatizar por completo esta tarea para eliminar los defectos que puedan haber aparecido en las piezas, con el consiguiente riesgo asociado.

Por último, otra de las mejoras identificadas se encuentra en el dispositivo de computación del sistema. Una buena opción es trasladar las operaciones de tratamiento de imagen al procesador de la tarjeta gráfica [VIT⁺12], disminuyendo así el tiempo de cómputo.

6.3.2 Mejorar el método de segmentación de imágenes

La segunda línea de trabajo identificada es la mejora del método de segmentación de las imágenes. En concreto, el método que hemos presentado en esta tesis doctoral (véanse los Capítulos 3 y 5) está basado en modelos previamente almacenados. En este sentido, existen otros enfoques no basados en conocimiento previo, que son más rápidos, y generalmente, consiguen detectar zonas potencialmente defectuosas empleando distintos filtros básicos de imagen [MFJ03], pero adolecen de una precisión inferior a los métodos basados en modelos.

6.3.3 Mejorar el método de categorización de defectos

La última de las líneas de trabajo futuro para esta tesis doctoral se basa en dos puntos. El primero de ellos hace referencia a las características extraídas para generar una representación de las regiones a categorizar. En el Capítulo 4 hemos realizado una evaluación completa compuesta por cuatro tipos distintos de características quedándonos, finalmente, con las que mejor han representado la información de las regiones. Una línea de investigación está relacionada con la búsqueda de nuevas características que ayuden a distinguir, de una forma más precisa las regiones defectuosas de las tolerables.

Por último, la segunda línea de investigación está relacionada con el método de categorización. Para esta tesis doctoral hemos empleado, en primera instancia, algoritmos de aprendizaje automático supervisado (véase el Capítulo 4), y posteriormente herramientas de clasificación colectiva con las que hemos reducido la cantidad de datos etiquetados (véase el Capítulo 5). En este sentido, la mejora vendría con la aplicación de técnicas de *Deep Learning* [KSB⁺10, FCNL12, KSH12], que han demostrado su validez con respecto a otros métodos.

6.4 Consideraciones finales

Los sistemas de visión artificial aplicados a la inspección de calidad tienen un arduo camino por delante en su tarea de complementar en la inspección manual. La complejidad inherente al proceso de control de calidad actual, con el conocimiento experto requerido para ella, hacen que actualmente no se disponga de un sistema sustitutivo de estas técnicas. Por todo ello, avanzar en este área supone un salto cualitativo, especialmente relevante para la industria, y que supone una importante reducción de costes y de tiempos.

En un mundo cada vez más competitivo y con unos bajos costes de producción, la calidad de los procesos productivos es la marca de diferenciación entre las diversas empresas. Por ende, hemos de asegurar que los productos manufacturados no sólo cumplen los estándares requeridos por la industria,

sino que la calidad tanto del proceso como del producto final va más allá, permitiendo el verdadero progreso de la sociedad. Y en este contexto, la automatización y la inteligencia artificial pueden ser un gran medio para ello. Al fin y al cabo,

«La pregunta de si los ordenadores pueden pensar es como la pregunta de si los submarinos pueden nadar.»

Edsger W. Dijkstra (1930–2002).

Bibliografía

- [A05] M. A. Survey on multiclass classification methods. *Neural networks*, páginas 1–9, 2005.
- [Ade83] F. Ade. Characterization of textures by eigenfilters. *Signal Processing*, 5(5):451–457, 1983.
- [AMSB99] S. Anand, C. McCord, R. Sharma, y T. Balachander. An integrated machine vision based system for solving the nonconvex cutting stock problem using genetic algorithms. *Journal of Manufacturing Systems*, 18(6):396–414, 1999.
- [ASS01] E. L. Allwein, R. E. Schapire, y Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *The Journal of Machine Learning Research*, 1:113–141, 2001.
- [AW99] S. Amari y S. Wu. Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, 12(6):783–789, 1999.
- [B⁺06] C.M. Bishop et al. *Pattern recognition and machine learning*. springer New York, 2006.
- [BAEG00] A. Baykut, A. Atalay, A. Erçil, y M. Güler. Real-time defect inspection of textured surfaces. *Real-Time Imaging*, 6(1):17–27, 2000.
- [BB03] M. Bennamoun y A. Bodnarova. Digital image processing techniques for automatic textile quality control. *Systems Analysis Modelling Simulation*, 43(11):1581–1614, 2003.
- [BBL02] A. Bodnarova, M. Bennamoun, y S. Latham. Optimal gabor filters for textile flaw detection. *Pattern Recognition*, 35(12):2973–2991, 2002.
- [BHR99] C. Bahlmann, G. Heidemann, y H. Ritter. Artificial neural networks for automated quality control of textile seams. *Pattern recognition*, 32(6):1049–1060, 1999.

- [BO01] K.L. Boyer y T. Ozguner. Robust online detection of pipeline corrosion from range data. *Machine Vision and Applications*, 12(6):291–304, 2001.
- [BP63] Mr. Bayes y M. Price. An essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, frs communicated by mr. price, in a letter to john canton, amfrs. *Philosophical Transactions (1683-1775)*, páginas 370–418, 1763.
- [Bra99] R. Bracewell. The fourier transform and its applications., 1999.
- [Bre01] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [BSPC05] R. Broadhurst, J. Stough, S. Pizer, y E. Chaney. Histogram statistics of local image regions for object segmentation. En *International Workshop on Deep Structure, Singularities, and Computer Vision*. Citeseer, 2005.
- [BW10] Z.M. Bi y L. Wang. Advances in 3d data acquisition and processing for industrial applications. *Robotics and Computer-Integrated Manufacturing*, 26(5):403–413, 2010.
- [BWBK97] A. Bodnarova, J.A. Williams, M. Bennamoun, y K.K. Kubik. Optimal textural features for flaw detection in textile materials. En *TENCON'97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications., Proceedings of IEEE*, tomo 1, páginas 307–310. IEEE, 1997.
- [Cam03] J. Campbell. *Castings*. Butterworth-Heinemann, 2003.
- [Cas96] K.R. Castleman. Digital image processing. second, 1996.
- [CBHK02] N.V. Chawla, K.W. Bowyer, L.O. Hall, y W.P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.
- [CGH97] E. Castillo, J.M. Gutiérrez, y A.S. Hadi. *Expert systems and probabilistic network models*. Springer Verlag, 1997.
- [CH91] G.F. Cooper y E. Herskovits. A bayesian method for constructing bayesian belief networks from databases. En *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, páginas 86–94. Morgan Kaufmann Publishers Inc., 1991.
- [CJ88] J. Chen y A.K. Jain. A structural approach to identify defects in textured images. En *Systems, Man, and Cybernetics, 1988. Proceedings of the 1988 IEEE International Conference on*, tomo 1, páginas 29–32. IEEE, 1988.

- [CP98] A. Conci y C.B. Proença. A fractal image analysis system for fabric inspection based on a box-counting method. *Computer Networks and ISDN Systems*, 30(20-21):1887–1895, 1998.
- [CP00] C. Chan y G.K.H. Pang. Fabric defect detection by fourier analysis. *Industry Applications, IEEE Transactions on*, 36(5):1267–1276, 2000.
- [CP02] A. Conci y C.B. Proença. A system for real-time fabric inspection and industrial decision. En *Proceedings of the 14th international conference on Software engineering and knowledge engineering*, páginas 707–714. ACM, 2002.
- [CPJT96] T.F. Cootes, G.J. Page, C.B. Jackson, y C.J. Taylor. Statistical grey-level models for object location and identification. *Image and Vision Computing*, 14(8):533–540, 1996.
- [CVGV⁺09] W. Cuypers, N. Van Gestel, A. Voet, J.P. Kruth, J. Mingneau, y P. Bleys. Optical measurement techniques for mobile and large-scale dimensional metrology. *Optics and Lasers in Engineering*, 47(3):292–300, 2009.
- [DB95] T. G. Dietterich y G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *arXiv preprint cs/9501101*, 1995.
- [DBKK03] E.P. DeGarmo, J.T. Black, R.A. Kohser, y B.E. Klamecki. Materials and process in manufacturing. *Jolm Wiley and Sons, USA*, página 974, 2003.
- [DKM65] J.V. Dawson, J.A. Kilshaw, y A.D. Morgan. The nature and origin of gas holes in iron castings. *AFS Transactions*, 73:224–240, 1965.
- [DP98] P.R. Drake y M.S. Packianather. A decision tree of neural networks for classifying images of wood veneer. *The International Journal of Advanced Manufacturing Technology*, 14(4):280–285, 1998.
- [DVAT82] R.L. De Valois, D.G. Albrecht, y L.G. Thorell. Spatial frequency selectivity of cells in macaque visual cortex. *Vision research*, 22(5):545–559, 1982.
- [FCNL12] C. Farabet, C. Couprie, L. Najman, y Y. LeCun. Learning hierarchical features for scene labeling. 2012.
- [FD11] Roman Filipovych y Christos Davatzikos. Semi-supervised pattern classification of medical images: Application to mild cognitive impairment (mci). *Neuroimage*, 55(3):1109–1119, 2011.

- [FGG97] N. Friedman, D. Geiger, y M. Goldszmidt. Bayesian network classifiers. *Machine learning*, 29(2):131–163, 1997.
- [FHJ52] E. Fix y J.L. Hodges Jr. Discriminatory analysis-nonparametric discrimination: Small sample performance. Informe técnico, DTIC Document, 1952.
- [G⁺95] S.R. Garner et al. Weka: The waikato environment for knowledge analysis. En *Proceedings of the New Zealand computer science research students conference*, páginas 57–64. Citeseer, 1995.
- [GR02] R.C. Gonzalez y E. Richard. Woods, digital image processing. ed: Prentice Hall Press, ISBN 0-201-18075-8, 2002.
- [HC04] Y. Huang y K.L. Chan. Texture decomposition by harmonics extraction from higher order statistics. *Image Processing, IEEE Transactions on*, 13(1):1–14, 2004.
- [Hei01] R.W. Heine. *Principles of metal casting*. McGraw-Hill, 2001.
- [HGT95] J.J. Hunter, J. Graham, y C.J. Taylor. User programmable visual inspection. *Image and vision computing*, 13(8):623–628, 1995.
- [HKY99] L. J. Heyer, S. Kruglyak, y S. Yooseph. Exploring expression data: identification and analysis of coexpressed genes. *Genome research*, 9(11):1106–1115, 1999.
- [HL02] C.W. Hsu y C.J. Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.
- [HPL⁺94] R.J. Hunsicker, J. Patten, A. Ledford, C. Ferman, M. Allen, y C. Ellis. Automatic vision inspection and measurement system for external screw threads. *Journal of Manufacturing Systems*, 13(5):370–384, 1994.
- [HSD73] R.M. Haralick, K. Shanmugam, y I.H. Dinstein. Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, 3(6):610–621, 1973.
- [HT98] T. Hastie y R. Tibshirani. Classification by pairwise coupling. *The annals of statistics*, 26(2):451–471, 1998.
- [IOV05] F. Isgro, F. Odone, y A. Verri. An open system for 3d data acquisition from multiple sensor. En *Computer Architecture for Machine Perception, 2005. CAMP 2005. Proceedings. Seventh International Workshop on*, páginas 52–57. IEEE, 2005.

- [IRV96] J. Iivarinen, J. Rauhamaa, y A. Visa. Unsupervised segmentation of surface defects. En *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, tomo 4, páginas 356–360. IEEE, 1996.
- [Jay80] G.S. Jayaraman. Spheroidization of graphite nodules in malleable iron. *Dissertation Abstracts International*, 40(12):209, 1980.
- [JCP00] A.R. Jimenez, R. Ceres, y J.L. Pons. A vision system based on a laser range-finder applied to robotic fruit harvesting. *Machine Vision and Applications*, 11(6):321–329, 2000.
- [JJCP99] A.R. Jimenez, A.K. Jain, R. Ceres, y J.L. Pons. Automatic fruit recognition: a survey and new results using range/attenuation images. *Pattern recognition*, 32(10):1719–1736, 1999.
- [JML00] J.Y. Jeng, T.F. Mau, y S.M. Leu. Gap inspection and alignment using a vision technique for laser butt joint welding. *The International Journal of Advanced Manufacturing Technology*, 16(3):212–216, 2000.
- [KCMP99] T.H. Kim, T.H. Cho, Y.S. Moon, y S.H. Park. Visual inspection system for the classification of solder joints. *Pattern Recognition*, 32(4):565–575, 1999.
- [KKS98] K.H. Kim, Y.W. Kim, y S.W. Suh. Automatic visual inspections system to detect wrongly attached components. En *Internat. Conf. on Signal Processing Applications and Technology*. 1998.
- [Kle96] E.M. Kleinberg. An overtraining-resistant stochastic modeling method for pattern recognition. *The annals of statistics*, 24(6):2319–2349, 1996.
- [KMM⁺94] J. Kittler, R. Marik, M. Mirmehdi, M. Petrou, y J. Song. Detection of defects in colour texture surfaces. *IAPR Machine Vision Applications*, páginas 558–567, 1994.
- [Kot72] E.L. Kotzin. *Metalcaster's reference and guide*. American Foundrymen's Society, 1972.
- [KP00] M. Kavoussanos y A. Pouliezios. Visionary automation of sack handling and emptying. *Robotics & Automation Magazine, IEEE*, 7(4):44–49, 2000.
- [KP02] A. Kumar y G.K.H. Pang. Defect detection in textured materials using optimized filters. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 32(5):553–570, 2002.

- [KSB⁺10] K. Kavukcuoglu, P. Sermanet, Y.L. Boureau, K. Gregor, M. Mathieu, y Y. LeCun. Learning convolutional feature hierarchies for visual recognition. *Advances in Neural Information Processing Systems*, páginas 1090–1098, 2010.
- [KSH12] A. Krizhevsky, I. Sutskever, y G. Hinton. Imagenet classification with deep convolutional neural networks. En *Advances in Neural Information Processing Systems 25*, páginas 1106–1114. 2012.
- [Kum00] V. Kumar. An introduction to cluster analysis for data mining. *Computer Science Department, University of Minnesota, USA*, 2000.
- [KZP07] S.B. Kotsiantis, I.D. Zaharakis, y PE. Pintelas. Supervised machine learning: A review of classification techniques. *Frontiers in Artificial Intelligence and Applications*, 160:3, 2007.
- [Lew98] D. Lewis. Naive (bayes) at forty: The independence assumption in information retrieval. *Machine Learning: ECML-98*, páginas 4–15, 1998.
- [Li09] S.Z. Li. *Markov random field modeling in image analysis*. Springer-Verlag New York Inc, 2009.
- [LJH63] C.R. Loper J. y R.W. Heine. The solidification of cast iron with spheroidal graphite. *Trans. ASM*, 56(1 s 135), 1963.
- [LSS⁺12] C. Laorden, B. Sanz, I. Santos, P. Galán-García, y P. G. Bringas. Collective classification for spam filtering. *Logic Journal of the IGPL*, página in press, 2012.
- [LZP05] L. Lei, X. Zhou, y M. Pan. Automated vision inspection system for the size measurement of workpieces. En *Instrumentation and Measurement Technology Conference, 2005. IMTC 2005. Proceedings of the IEEE*, tomo 2, páginas 872–877. IEEE, 2005.
- [Man83] B.B. Mandelbrot. *The fractal geometry of nature*. Wh Freeman, 1983.
- [MEDT96] M. Moganti, F. Ercal, C.H. Dagli, y S. Tsunekawa. Automatic pcb inspection algorithms: a survey. *Computer Vision and Image Understanding*, 63(2):287–313, 1996.
- [Mer03] D. Mery. Crossing line profile: a new approach to detecting defects in aluminium die casting. *Image Analysis*, páginas 245–256, 2003.
- [MFJ03] D. Mery, D. Filbert, y T. Jaeger. Image processing for fault detection in aluminum castings. 2003.

- [MGD00] B. Mallik-Goswami y A.K. Datta. Detecting defects in fabric with laser-based morphological image processing. *Textile Research Journal*, 70(9):758–762, 2000.
- [MGS98] A. Mital, M. Govindaraju, y B. Subramani. A comparison between manual and hybrid methods in parts inspection. *Integrated Manufacturing Systems*, 9(6):344–349, 1998.
- [MMT04] A. Monadjemi, M. Mirmehdi, y B. Thomas. Restructured eigenfilter matching for novelty detection in random textures. *learning*, 5:13, 2004.
- [Mon04] A. Monadjemi. *Towards efficient texture classification and abnormality detection*. Tesis Doctoral, University of Bristol, 2004.
- [MP05] T. Mäenpää y M. Pietikäinen. Texture analysis with local binary patterns. *Handbook of Pattern Recognition and Computer Vision*, 3:197–216, 2005.
- [MPZ⁺03] E.N. Malamas, E.G.M. Petrakis, M. Zervakis, L. Petit, y J.D. Legat. A survey on industrial vision systems, applications and tools. *Image and vision computing*, 21(2):171–188, 2003.
- [MS95] M. Magee y S. Seida. An industrial model based computer vision system. *Journal of Manufacturing Systems*, 14(3):169–186, 1995.
- [MTP03] T. Mäenpää, M. Turtinen, y M. Pietikäinen. Real-time surface inspection by texture. *Real-Time Imaging*, 9(5):289–296, 2003.
- [NSBG09] G. Namata, P. Sen, M. Bilgic, y L. Getoor. Collective classification for text classification. *Text Mining*, páginas 51–69, 2009.
- [OPH96] T. Ojala, M. Pietikäinen, y D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.
- [OPM02] T. Ojala, M. Pietikainen, y T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):971–987, 2002.
- [Par71] W.B. Parkes. *Clay-bonded foundry sand*. Applied science publishers, 1971.
- [Per04] F. Pernkopf. 3d surface inspection using coupled hmms. En *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, tomo 3, páginas 223–226. IEEE, 2004.
- [Pfa06] B. Pfahringer. A semi-supervised spam mail detector. 2006.

- [PMV02] M. Pietikainen, T. Maenpaa, y J. Viertola. Color texture classification with color histograms and local binary patterns. En *Workshop on Texture Analysis in Machine Vision*, páginas 109–112. Citeseer, 2002.
- [PO02] F. Pernkopf y P. O’Leary. Visual inspection of machined metallic high-precision surfaces. *EURASIP Journal on Applied Signal Processing*, 2002(1):667–678, 2002.
- [PO03] F. Pernkopf y P. O’Leary. Image acquisition techniques for automatic visual inspection of metallic surfaces. *NDT & E International*, 36(8):609–617, 2003.
- [Qui86] J.R. Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.
- [Qui93] J.R. Quinlan. *C4. 5: programs for machine learning*, tomo 1. Morgan kaufmann, 1993.
- [Rah08] M. Rahayem. An industrial robot as part of an automatic system for geometric reverse engineering. *Robot manipulators*, 2008.
- [Rao00] P.N. Rao. *Manufacturing technology: metal cutting and machine tools*. Tata McGraw-Hill, 2000.
- [Sch05] J. Scharcanski. Stochastic texture analysis for monitoring stochastic processes in industry. *Pattern recognition letters*, 26(11):1701–1709, 2005.
- [SFF10] R. Socher y L. Fei-Fei. Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. En *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, páginas 966–973. IEEE, 2010.
- [Sha51] C.E. Shannon. Prediction and entropy of printed english. *Bell System Technical Journal*, 30(1):50–64, 1951.
- [SJ99] Y.R. Shiau y B.C. Jiang. Study of a measurement algorithm and the measurement loss in machine vision metrology. *Journal of manufacturing systems*, 18(1):22–34, 1999.
- [SKM09] Y. Singh, A. Kaur, y R. Malhotra. Comparative analysis of regression and machine learning methods for predicting fault proneness models. *International Journal of Computer Applications in Technology*, 35(2):183–193, 2009.
- [SLTW04] V. Svetnik, A. Liaw, C. Tong, y T. Wang. Application of breiman’s random forest to modeling structure-activity relationships of pharmaceutical molecules. *Multiple Classifier Systems*, páginas 334–343, 2004.

- [SPK97] L. Shafarenko, M. Petrou, y J. Kittler. Automatic watershed segmentation of randomly textured color images. *Image Processing, IEEE Transactions on*, 6(11):1530–1544, 1997.
- [SQ10] Z. Shunli y Y. Qingshuang. Personal spam filter by semi-supervised learning. 2010.
- [SRR⁺03] W. Scott, G. Roth, J.F. Rivest, et al. View planning for automated 3d object reconstruction inspection. *ACM Computing Surveys*, 35(1), 2003.
- [SSH99] M.L. Smith, G. Smith, y T. Hill. Gradient space analysis of surface defects using a photometric stereo derived bump map. *Image and vision computing*, 17(3-4):321–332, 1999.
- [STD09] G. Sansoni, M. Trebeschi, y F. Docchio. State-of-the-art and applications of 3d imaging sensors in industry, cultural heritage, medicine, and criminal investigation. *Sensors*, 9(1):568–601, 2009.
- [SUPS⁺11] S. Santos, X. Ugarte-Pedrero, B. Sanz, C. Laorden, y P. G. Bringas. Collective classification for packed executable identification. En *Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, páginas 23–30. 2011.
- [TCC98] D.M. Tsai, J.J. Chen, y J.F. Chen. A vision system for surface roughness assessment using neural networks. *The International Journal of Advanced Manufacturing Technology*, 14(6):412–422, 1998.
- [TH99] D.M. Tsai y C.Y. Hsieh. Automated surface inspection for directional textures. *Image and Vision Computing*, 18(1):49–62, 1999.
- [TH03] D.M. Tsai y T.Y. Huang. Automated surface inspection for statistical textures. *Image and Vision Computing*, 21(4):307–323, 2003.
- [TL08] F. Truchetet y O. Laligant. Review of industrial applications of wavelet and multiresolution-based signal and image processing. *Journal of Electronic Imaging*, 17:031102, 2008.
- [TSA⁺98] F. Torres, J.M. Sebastian, R. Aracil, L.M. Jimenez, y O. Reinoso. Automated real-time visual inspection system for high-resolution superimposed printings. *Image and vision computing*, 16(12-13):947–958, 1998.
- [Tsa99] D.M. Tsai. A machine vision approach for detecting and inspecting circular parts. *The International Journal of Advanced Manufacturing Technology*, 15(3):217–221, 1999.

- [TSI97] H.K. Toenshoff, C. Soehner, y G. Isensee. Vision-guided tripod material transport system for the packaging industry. *Robotics and Computer-Integrated Manufacturing*, 13(1):1–7, 1997.
- [ÜMB06] B. Üstün, W.J. Melssen, y L.M.C. Buydens. Facilitating the application of support vector regression by using a universal pearson vii function based kernel. *Chemometrics and Intelligent Laboratory Systems*, 81(1):29–40, 2006.
- [UPSGB11] X. Ugarte-Pedrero, I. Santos, y P.G. Bringas. Semi-supervised learning for packed executable detection. En *In Proceedings of the 7th International Conference on Information Security and Cryptology (INSCRYPT)*. 2011. In press.
- [VIT⁺12] S.N. Swetadri Vasan, C.N. Ionita, A.H. Titus, A.N. Cartwright, D.R. Bednarek, y S. Rudin. Graphics processing unit (gpu) implementation of image processing algorithms to improve system performance of the control, acquisition, processing, and image display system (capids) of the micro-angiographic fluoroscope (maf). En *Proc. of SPIE Vol*, tomo 8313, páginas 83134C–1. 2012.
- [VJ04] P. Viola y M.J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.
- [VKM99] J. Velten, A. Kummert, y D. Maiwald. Real time railroad tie inspection implemented on dsp and fpga boards. En *International Conference on Signal Processing Applications and Technology (ICSPAT'99)*. 1999.
- [VNP86] F.M. Vilnrotter, R. Nevatia, y K.E. Price. Structural analysis of natural textures. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (1):76–89, 1986.
- [VS07] D. Vom Stein. Automatic visual 3-d inspection of castings. *Foundry Trade Journal*, 180(3641):24–27, 2007.
- [WKC10] J. Wang, S. Kumar, y S.F. Chang. Semi-supervised hashing for scalable image retrieval. En *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, páginas 3424–3431. IEEE, 2010.
- [Woo90] E.J. Wood. Applying fourier and associated transforms to pattern characterization in textiles. *Textile Research Journal*, 60(4):212–220, 1990.
- [WX99] W. Wen y A. Xia. Verifying edges for visual inspection purposes. *Pattern recognition letters*, 20(3):315–328, 1999.

- [XHY09] X. Xiang, J. He, y S. Yang. Pinhole defects detection of aluminum foil based on machine vision. En *Electronic Measurement & Instruments, 2009. ICEMI'09. 9th International Conference on*, páginas 2–38. IEEE, 2009.
- [Xie08] X. Xie. A review of recent advances in surface defect detection using texture analysis techniques. *ELCVIA*, 7(3), 2008.
- [XM05a] X. Xie y M. Mirmehdi. Localising surface defects in random colour textures using multiscale texem analysis in image eigenchannels. En *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, tomo 3, páginas III–1124. IEEE, 2005.
- [XM05b] X. Xie y M. Mirmehdi. Texture exemplars for defect detection on random textures. *Pattern Recognition and Image Analysis*, páginas 404–413, 2005.
- [XM07] X. Xie y M. Mirmehdi. Texems: Texture exemplars for defect detection on random textured surfaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(8):1454–1464, 2007.
- [XMT06] X. Xie, M. Mirmehdi, y B. Thomas. Colour tonality inspection using eigenspace features. *Machine Vision and Applications*, 16(6):364–373, 2006.
- [YPY05] X. Yang, G. Pang, y N. Yung. Robust fabric defect detection and classification using multiple adaptive wavelets. En *Vision, Image and Signal Processing, IEE Proceedings-*, tomo 152, páginas 715–723. IET, 2005.

