



UNIVERSIDAD CATÓLICA DEL NORTE  
FACULTAD DE INGENIERÍA Y CIENCIAS GEOLÓGICAS  
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

**CLASIFICACIÓN DE MINERAL DE SOBRE TAMAÑO  
EN PROCESOS MINEROS UTILIZANDO  
DESCRIPTORES DE TEXTURA SOBRE IMÁGENES  
DIGITALES**

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN INGENIERÍA  
INFORMÁTICA

Adrian Camilo Pupo Urrutia  
Profesor guía:  
Dr. Juan Bekios Calfa

Antofagasta, Chile  
Diciembre, 2018

# ÍNDICE GENERAL

ÍNDICE DE FIGURAS . . . . .	viii
NOMENCLATURA . . . . .	ix
RESUMEN . . . . .	x
<b>I. Introducción . . . . .</b>	<b>1</b>
1.1. Planteamiento del problema . . . . .	1
1.2. Descripción formal del problema . . . . .	3
1.3. Escenario de aplicación . . . . .	4
1.4. Descripción de la solución propuesta . . . . .	6
1.5. Motivación . . . . .	7
1.6. Pregunta de Hipótesis . . . . .	8
1.7. Hipótesis de Trabajo . . . . .	8
1.8. Objetivos . . . . .	8
1.8.1. Principales contribuciones . . . . .	9

1.9. Estructura de la tesis . . . . .	9
<b>II. Marco Teórico . . . . .</b>	<b>11</b>
2.1. Introducción . . . . .	11
2.2. Reconocimiento de Imágenes. . . . .	11
2.3. Análisis de Imagen. . . . .	13
2.3.1. Tipos de características a extraer de una Imagen . . . . .	13
2.3.2. Descriptores . . . . .	14
2.3.2.1. Patrón Local Binario (LBP) . . . . .	15
2.3.2.2. Histograma de Gradiente Orientados (HOG) . . . . .	23
2.3.3. Detección del objeto . . . . .	33
2.4. Aprendizaje Automático . . . . .	34
2.4.1. Aprendizaje Supervisado . . . . .	35
2.4.1.1. Máquina de Soporte Vectores . . . . .	36
2.5. Evaluación del rendimiento de un clasificador . . . . .	42
2.6. Métricas . . . . .	43
2.6.1. Matriz de confusión Multiclase . . . . .	44
2.6.2. Curvas Roc . . . . .	47
<b>III. Estado del arte . . . . .</b>	<b>50</b>
<b>IV. Metodología . . . . .</b>	<b>56</b>

4.1. Metodología . . . . .	56
4.2. Introducción . . . . .	56
4.3. Adquisición y clasificación de la Imagen . . . . .	57
4.4. Procesamiento de las Imágenes . . . . .	58
4.5. Extracción de características de las Imágenes . . . . .	59
4.6. Clasificación de imágenes . . . . .	61
4.7. Validación de los resultados . . . . .	63
<b>V. Resultados Experimentales . . . . .</b>	<b>65</b>
5.1. Descriptor LBP . . . . .	65
5.1.1. Configuración del experimento . . . . .	66
5.1.2. Validación de los Resultados . . . . .	66
5.2. Descriptor HOG . . . . .	68
5.2.1. Configuración del experimento . . . . .	68
5.2.2. Validación de los Resultados . . . . .	69
5.3. Evaluación del rendimiento de los modelos LBP y HOG . . . . .	70
5.3.1. Eficiencia de los clasificadores . . . . .	71
5.3.2. Eficiencia computacional . . . . .	73
<b>VI. Conclusiones y trabajos futuros . . . . .</b>	<b>75</b>
6.1. Conclusiones . . . . .	75

6.2. Trabajos Futuros . . . . .	77
<b>Apéndice A. Publicaciones . . . . .</b>	<b>78</b>
<b>BIBLIOGRAFÍA . . . . .</b>	<b>88</b>

# ÍNDICE DE FIGURAS

	<b>Página</b>
1.1. Diagrama de flujo procesamiento de reducción de Minerales. . . . .	2
1.2. Escenario de la Aplicación. . . . .	4
1.3. Diagrama de flujo procesamiento de reducción de Minerales. . . . .	7
2.1. Análisis representativo del procesamiento de la Imagen. . . . .	12
2.2. Funcionamiento del operador LBP. . . . .	16
2.3. Distribución de los vecinos para LBP según su Radio. . . . .	17
2.4. Distribución de los vecinos para LBP según cantidad de puntos. . . . .	17
2.5. Histograma de ejemplo utilizando 13 Bins . . . . .	18
2.6. Representación visual de los valores LBP con 8 vecinos y radio 1 . . . . .	18
2.7. Ejemplo de dos patrones iguales pero con diferente orientación . . . . .	19
2.8. Ejemplo de todas las posibles rotaciones de un vector y su valor de LBP. . . . .	19
2.9. Patrones uniformes y no uniformes en LBP. . . . .	20
2.10. Representación de texturas definidas LBP . . . . .	20

2.11. Representación de un mismo histograma LBP para imágenes distintas. . . . .	21
2.12. Ejemplo de aplicación errónea de histogramas sin bloques . . . . .	21
2.13. División de la imagen I de N bloques. . . . .	22
2.14. Histogramas individuales de los N bloques. . . . .	22
2.15. Intercalar rejillas de mallado de N bloques. . . . .	22
2.16. Histogramas Normalizado de N bloques. . . . .	23
2.17. Proceso para el calculo del descriptor HOG. . . . .	24
2.18. Ejemplo para el cálculo de los gradientes direccionales en un pixel dado. . . . .	26
2.19. Cálculo de las orientaciones y las magnitudes en un píxel dado. . . . .	26
2.20. División de las celdas de tamaño fijos de píxeles. . . . .	27
2.21. Histograma de orientaciones de gradientes en cada una de las celdas. . . . .	28
2.22. Diagrama de orientaciones e intervalos para los gradientes. . . . .	29
2.23. Acumulación de la magnitud del gradiente. . . . .	29
2.24. Histograma de cada una de las celdas de un bloque determinado. . . . .	30
2.25. Histograma y Vector de características. . . . .	31
2.26. La representación normalizada . . . . .	32
2.27. Contribución de las celdas a la descripción de varios bloques. . . . .	32
2.28. Hiperplano de separación de dos clases. . . . .	37
2.29. Separación de datos mediante SVM. . . . .	37
2.30. SVM linealmente separable. . . . .	38

2.31. SVM con margen máximo . . . . .	40
2.32. Conjunto de datos linealmente no separables . . . . .	40
2.33. Transformación de datos de entrada a un espacio de dimensión mayor . . . . .	41
2.34. SVM no lineal inducida por una función Kernel . . . . .	41
2.35. Ejemplo del comportamiento de la curva Roc. . . . .	48
4.1. Pasos principales desde la adquisición de imágenes hasta la clasificación de rocas .	57
4.2. Clases de minerales ocupados en el proceso. . . . .	58
4.3. Método de estimación basado en $k$ rodajas. . . . .	63
5.1. Curva ROC para la clase "Rocas Finas" . . . . .	71
5.2. Curva ROC para la clase "Rocas Medianas" . . . . .	72
5.3. Curva ROC para la clase "Rocas Finas-Medianas" . . . . .	73
5.4. Curva ROC para la clase "Rocas Medianas-Gruesas" . . . . .	74

# NOMENCLATURA

**AUC** Es el área formada bajo la curva ROC (*Area Under the Curve*).

**Ceduc UCN** Base de datos privada de imágenes de mineral de propiedad del Ceduc UCN. Antofagasta, Chile.

**Curva ROC** Representación gráfica bidimensional de las tasa de verdaderos positivos versus los falsos positivos.

**FOLD** Término utilizado cuando se realiza validación cruzada de datos *K-Fold* (*K-Fold cross-validation*).

**HOG** Histograma de Gradientes orientados (*Histogram of Oriented Gradients*).

**LBP** Patrones binarios locales (*Local binary patterns*).

**OpenCV** Librería libre para visión por computador.

**RGB** Representación de un color mediante la mezcla de tres colores primarios Rojo, Verde y Azul (*Red, Green, Blue*).

**SVM** Máquina de soporte de vectores (*Support Vector Machine*).

# RESUMEN

En la actualidad la industria del cobre se encuentra mejorando los procesos que involucran gran consumo energético, como lo es el chancado y la molienda en donde el tamaño de la roca es un factor importante en esta área. Si bien es cierto, se han ido diseñando nuevas alternativas para la detección del tamaño de las rocas basado en imágenes, siguen existiendo pequeños detalles para complementar dichos avances. En relación a lo anterior esta propuesta de tesis plantea el estudio de algoritmos para la detección de rocas que se encuentren con sobretamaño dentro del proceso de producción.

Esto se llevó acabo utilizando la extracción de características de una base de datos de imágenes de rocas divididas en cuatro categorías, extrayendo por cada grupo de rocas los descriptores HOG (Histograma de Gradiente Orientados) y LBP (Patrón Local Binario). Al mismo tiempo se revisaron los diferentes factores que alteran o modifican la forma en que se detectan las rocas como lo pueden ser el ambiente o la escena en que se movilizan. Dichos descriptores fueron evaluados por un clasificador a través de estudios de algoritmos de aprendizaje supervisado como lo es SVM (Support Vector Machine) para identificar posibles errores y realizar su corrección, cuyo fin fue otorgar nuevos métodos para la detección de rocas con sobre-tamaño.

Se comprobó que a pesar que el descriptor LBP tiene estudios para realizar este tipo de desarrollo, como es la detección y clasificación de minerales por medio de la textura en imágenes 2D. El descriptor HOG propuesto es el más confiable indiscutiblemente en precisión frente al LBP, debido que este tiene un 96,1 % frente al LBP con un 90,8 %. Presentando un buen desempeño en casi todos los distintos tipos de minerales a clasificar.

En resumen, la tesis demuestra experimentalmente que los descriptores HOG tienen los mejores resultados de rendimiento en este tipo de imágenes.

# Capítulo I

## Introducción

### 1.1. Planteamiento del problema

Hoy en día el norte de Chile cuenta con una gran cantidad de yacimientos de cobre que requieren nuevas tecnologías que ayuden a los procesos productivos. Es sabido de ante mano que el proceso de extracción y producción del mineral consta de distintas etapas, ver Figura 1.1, donde cada una de ellas es de suma importancia para la óptima producción del mineral.

La etapa de interés es la del proceso de chancado y molienda. Estos procesos de tipo mecánico determinan la distribución del tamaño de las rocas fragmentadas en las diversas etapas de la trituración hasta el nivel de polvo. Estos procesos, en general, tienen un gran consumo de energía y es un factor crítico para las empresas no solo por el consumo de energía eléctrica sino también porque tiene normas legales a las cuales debe regirse como lo es la ISO 50001 y que tiene como objetivo dar una estructura en los sistemas y procesos que sean necesarios para mejorar el desempeño energético, donde se incluye la eficiencia energética y los insumos en cada uno de los procesos o áreas de la empresa.

Cabe destacar que los minerales brutos extraídos hace años contaban con mayor concentración de mineral (Ley) que los actuales en comparación con la minería que tiene un proceso continuo o de años, como las actuales y por ello los tamaños de liberación eran superiores. Hoy en día, la falta de estos minerales con alta ley ha llevado a la reducción de tamaño para la liberación obtenido en el proceso de la molienda. Por ello, predecir el nivel y el grado de fragmentación del mineral es importante para la industria, ya que involucra la reducción significativa de los costos de trituración

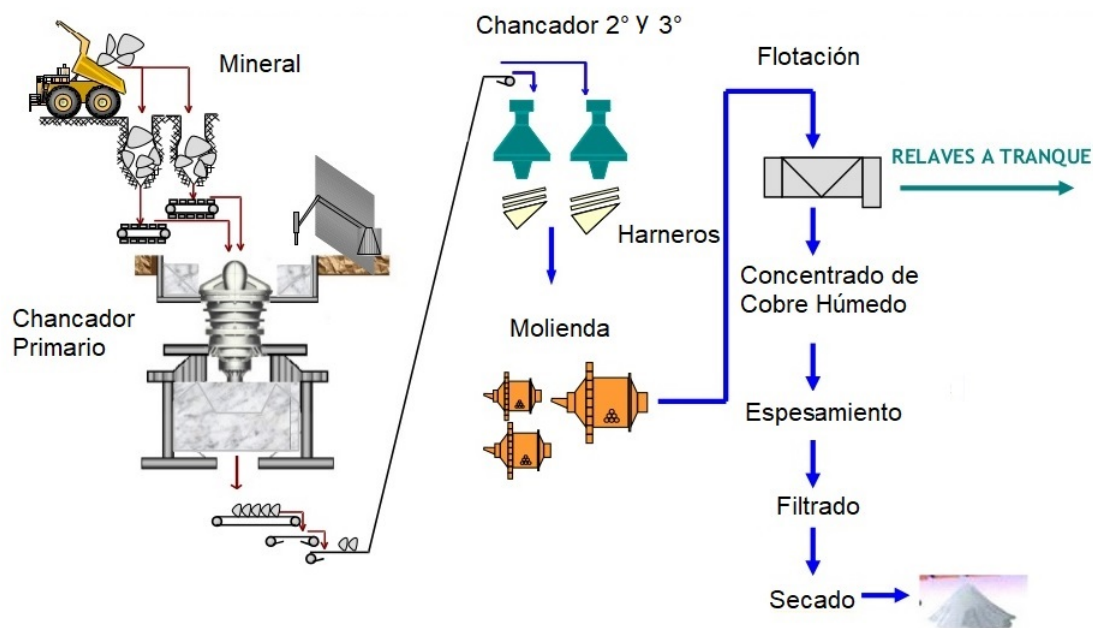


FIGURA 1.1: Diagrama de flujo procesamiento de reducción de Minerales.

y molienda mejorando la eficiencia del producto.

Las mayores anomalías en las cuales se ven afectados estos procesos están asociadas a las características del mineral de alimentación como lo es la dureza, distribución de tamaño y densidad. Debido a estos la distribución del tamaño del mineral juega un papel fundamental [1].

Cuando se trabaja con una distribución de tamaño anormal, material de sobretamaño o mineral grueso, el proceso de chancado sufre un aumento en el consumo eléctrico lo que puede dañar los bujes o partes internas del equipo debido al aumento de temperatura [2].

La molienda se encarga de reducir de tamaño aquellas partículas de las cuales está compuesta un mineral provenientes del chancado secundario y terciario, donde sin lugar a duda la distribución del tamaño o granulometría provenientes de los chancadores influyen en su rendimiento. La granulometría, en general, en la etapa de molienda es de mayor incidencia en rendimiento energético y metalúrgico.

En resumen es necesario que se cumplan tres condiciones mínimas para la operación estable de un molino, y no generar gastos energéticos innecesarios dentro del proceso:

1. Mantener constante el tonelaje procesado y la calidad del mineral de molienda.

2. Mantener el tamaño del mineral de la molienda constante.
3. Maximizar el tonelaje procesado manteniendo la calidad del tamaño de molienda dentro de límites específicos.

Se debe tener en cuenta que la falta o ausencia de estas condiciones no solo pueden generar aumentos de energía eléctrica en el sistema, si no desgastes en los revestimientos del molino, aumento en el nivel de llenado de bolas entre otros. Es por ello que la clasificación del mineral en la molienda juega un papel importante para obtener un producto en condiciones de operación y eficiencia.

Para solucionar el problema de clasificación de mineral existe un sin número de métodos que permiten estimar la distribución del tamaño de las rocas, siendo el tamizado o cribado el más popular y antiguo. Está ampliamente reconocido que la popularidad de tamizado se debe a su simplicidad y costo relativamente bajo. Sin embargo, el proceso puede ser tedioso y consume mucho tiempo particularmente con el manejo de grandes cantidades de rocas. El alto costo de su implementación ha llegado a desarrollar métodos indirectos para estimar las distribuciones de tamaño. Uno de los más populares es el análisis de fotografías, donde las imágenes se obtienen de las rocas en pilas o correas transportadoras sea a mano o por medio de técnicas de procesamiento de imágenes. Por lo tanto, la necesidad de analizar el tamaño de rocas fue una de las principales motivaciones para la introducción de un sistema para la clasificación de tamaño de rocas, basadas en imágenes en la industria minera.

## **1.2. Descripción formal del problema**

El problema consiste en la identificación de un descriptor que permita una clasificación de los distintos tipos de minerales por medio de identificar todas las regiones de la imagen que permitan la extracción de sus propiedades de interés, como los bordes, iluminación, texturas, manchas, terminales de líneas entre otros; mediante el análisis de imágenes digitales. La calidad en la clasificación de los minerales es un punto primordial en la construcción de la solución, por la cual está debe ser cuantificable. Para ello se realizó una comparación que precise el comportamiento efectivo de la calidad de falsos positivos y falsos negativos encontrados en la detección de la clasificación del mineral. Esta comparación se realizó evaluando el rendimiento de los descriptores utilizando métricas de validación que se emplean en el aprendizaje supervisado.

En el presente trabajo, se estudiará cómo utilizar los descriptores de texturas para la clasificación de imágenes de mineral en diferentes tamaños. El proceso comprende un primer paso de extracción de características por medio de los descriptores estudiados en esta tesis. El uso de un clasificador basado en técnicas de aprendizaje automático como SVM (Máquina de vectores de soporte). Y finalmente, la validación de los resultados por el medio de un método que defina las condiciones de rendimiento de cada una de las técnicas utilizadas.

### 1.3. Escenario de aplicación

Debido a la naturaleza experimental de este trabajo, el desarrollo de los modelos propuestos fue implementado en el lenguaje de programación *Python* y se utilizó la IDE (*Integrated Development Environment*) *Spyder* (*Scientific Python Development Environment*), ver 1.2, para el diseño y pruebas de los experimentos de la tesis.

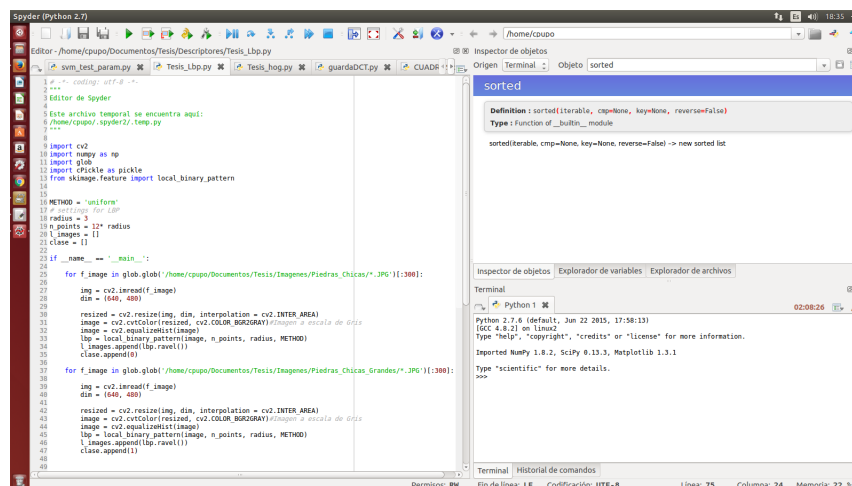


FIGURA 1.2: Escenario de la Aplicación.

La elección del software fue debido a sus funciones avanzadas de edición, entorno informático numérico, entorno de desarrollo de código abierto (IDE) para programaciones científicas como lo es *Python*).

**Python:** Es un lenguaje de propósito general ampliamente usado para el análisis de datos.

Este lenguaje se usa frecuentemente para el procesado de información, creación de plataformas *webs* y *machine learning*. Todas estas funcionalidades las aportan el gran número de librerías a su disposición. *Python* al contar con una gran comunidad tiene un amplio

soporte, esto le proporciona un gran número de librerías *open source* [3]. Los implementados fueron:

- *Pickle*: El módulo *pickle* implementa un algoritmo básico que nos permite guardar un objeto a disco para poder recuperarlo más tarde, o puede que nos sea necesario enviar un objeto a través de la red. Lo cual es definido como el acto de poder convertir objetos a cadenas de bytes. [4].
- *Glob*: Nos permite examinar el contenido el cual se encuentra en determinado directorio, para extraer archivos como carpetas y filtros de los contenidos necesarios.
- *OpenCV* : Es una librería la cual fue creada por Intel para analizar y procesar imágenes por computador, la cual está disponible en múltiples plataformas como: Windows, Linux, Mac, Android. Al igual cuenta con soporte para los distintos lenguajes de programación como: Python, Java, C/C++, .Net, entre otros [5].

Para la implementación del algoritmo de aprendizaje automático, se han utilizado librerías especializadas en este propósito, estas librerías son capaces de trabajar con estructuras específicas de datos, y funcionalidades específicas de *machine learning*.

**Numpy:** Es la librería por excelencia para cualquier análisis de datos, esta librería permite el manejo de estructuras de datos de una manera eficiente. Esta librería está implementada en C, gracias a esto consigue optimizar procesos como, los accesos a memoria y el tiempo de cómputo, evitando así sufrir la lentitud que los lenguajes interpretados como *Python* incluyen inherentemente.

*Numpy* ante todo proporciona estructuras para encapsular los datos, como arreglos o matrices n-dimensionales implementados de manera propia, para evitar así tener que utilizar *Python* mucho más lentos, además de poseer una gran cantidad de funciones para el manejo de dichas estructuras [3].

**Scipy:** Es una biblioteca de funciones matemáticas de alto nivel, esto permite trabajar con toda clase de datos de una manera muy potente, se llega a comparar con otras herramientas matemáticas como *Matlab* y *Octave*. Contiene un gran número de operaciones de optimización, álgebra lineal y procesamiento de señales. El aspecto de *Scipy* más útil para este proyecto es la gran compatibilidad que tiene con el resto de herramientas, ya que ofrece soporte para varias funciones de *Numpy*, y para las librerías de *machine learning* [3].

**Scikit-image:** Es una biblioteca de procesamiento de imágenes que implementa algoritmos y utilidades para su uso en aplicaciones de investigación, educación e industria. Se distribuye bajo la licencia BSD de código abierto modificado, proporciona una API bien documentada en el lenguaje de programación Python y está desarrollada por un equipo activo e internacional de colaboradores [6]. De las cuales ocupamos los descriptores:

- HOG: Nos permite extraer los Histogramas de gradientes de orientación para una determinada imagen [7].
- LBP: Esta función nos permite extraer las texturas de una imagen [8].

**Scikit-Learn:** Es una librería especializada en *machine learning*, y una tecnología fundamental en el campo del aprendizaje máquina. Esta librería cubre ampliamente todas las necesidades básicas que se puedan necesitar en una herramienta de aprendizaje automático, tanto algoritmos de aprendizaje, como funciones que proveen diversas utilidades, como, por ejemplo, obtener los mejores parámetros para la creación de un modelo o clases para obtener métricas sobre el rendimiento de tu modelo [3]

Las funciones que hemos empleado de esta librería son:

- SVC: Es nuestro estimador, se encarga de crear nuestro modelo, en función de los parámetros que elijamos para ello, se basa en el funcionamiento del algoritmo “máquinas de vector soporte” enfocadas a la clasificación [3].
- GridSearchCV: Esta función encuentra los mejores parámetros para un estimador dado, de manera que podemos construir un modelo lo más exacto posible. Este método funciona de la siguiente manera: crea un *grid* con todas las combinaciones de argumentos posibles y se las pasa al estimador para que cree el modelo, y así evaluar su precisión. Es una función que ahorra varios procesos, y lo sustituye por una función bastante intuitiva [3]

## 1.4. Descripción de la solución propuesta

El trabajo comienza con la creación de una base de datos de imágenes en dos dimensiones, las cuales están divididas en cuatro categorías de minerales (Finas, Finas-Mediadas, Mediadas-grandes, Grandes), permitiendo contar con los minerales de referencia para clasificar en el proceso. Posteriormente estas imágenes se reducirán de tamaño y se implementarán los

descriptores HOG y LBP por cada tamaño seleccionado. A continuación se procederá a aplicar el clasificador SVM para cada uno de los descriptores generados por cada tamaño de la imagen de las cuatro categorías y comprobando sus resultados. Por último, se utiliza una matriz de confusión para proporcionar la información necesaria para evaluar y comprobar si el descriptor HOG tiene mejor precisión que el LBP o viceversa, mostrada en la Figura 1.3

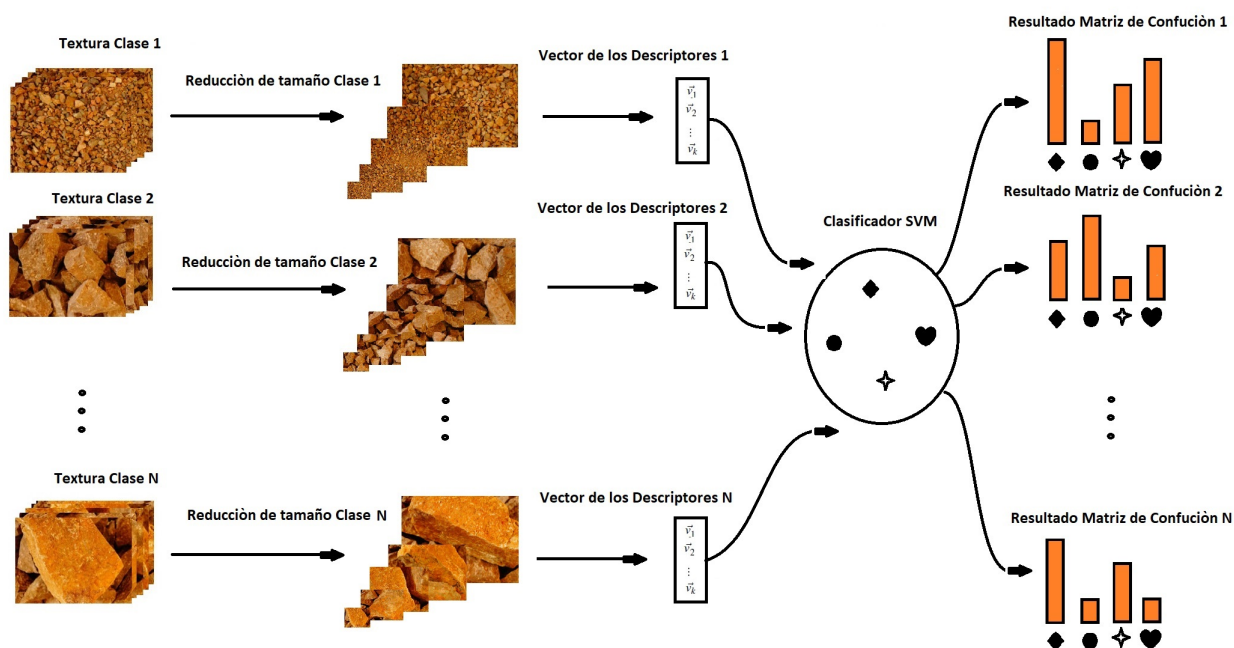


FIGURA 1.3: Diagrama de flujo procesamiento de reducción de Minerales.

## 1.5. Motivación

Hoy en día unos de los principales problemas que presenta la industria minera en los procesos de chancado y molienda es el intenso gasto de energía eléctrica, por ello la variación de tamaño y dureza del mineral tienen un impacto directo sobre éste. Es por ello que la clasificación de las rocas antes que pase por estos procesos juega un papel muy importante para reducirlo.

Como resultado se realizó un estudio significativo para medir o estimar la distribución de tamaño del material particulado con el fin de evaluar y optimizar la producción en términos de tasa y costos de producción (Energía, materiales y costos de equipo). Los operadores de minas desean medir el tamaño de las partículas de los minerales pero en la actualidad la distribución o clasificación de tamaños de rocas más usado en el mundo trabaja bajo un dimensionamiento mecánico, donde

métodos como tamizado o cribado no proporcionan información que posteriormente pueda ser utilizados para el control en línea del proceso, al igual son herramientas de evaluación imperfectas debido a la retroalimentación lenta, y la medición inconsistente. Es por ello que la detección de objetos en el procesamiento de imágenes fue el área de interés por los avances presentados en la clasificación de minerales en procesos industriales y de laboratorio buscando así nuevas soluciones para la mejora continua del proceso [9, 10, 11, 12]. El Determinar en tiempo real la dureza y el tamaño de las rocas puede mejorar el rendimiento del molino y el consumo de energía [13] ya que el 55% de la energía total requerida en una operación minera se utiliza en el proceso de la molienda.

En este trabajo, se han estudiado y comparado los descriptores HOG y LBP para la clasificación de minerales, debido que han presentado estudios relevantes que nos permitieron tener resultados óptimos para el proceso [14, 15, 16].

## **1.6. Pregunta de Hipótesis**

Esta tesis intentará responder la siguiente pregunta:

¿La implementación de descriptores de textura como HOG o LBP para la clasificación granulométrica de minerales utilizando imágenes digitales mejora su eficiencia en términos de tasa de acierto con respecto a otras técnicas aplicadas para el mismo propósito?

## **1.7. Hipótesis de Trabajo**

$H_1$  La clasificación granulométrica de minerales utilizando imágenes digitales mejora en términos de tasa de acierto cuando se utilizan descriptores de textura.

## **1.8. Objetivos**

Definida la hipótesis, a continuación se plantean el objetivo general y los específicos, con los cuales se podrá medir el avance de la investigación.

## **Objetivo general**

Comparar la eficiencia y el índice de rendimiento del descriptor HOG con respecto al descriptor LBP para la clasificación de rocas con sobre-tamaño.

## **Objetivo específicos**

1. Revisar el estado del arte de los sistemas para el procesamiento de imágenes de rocas para la estimación de sobre tamaño.
2. Generar una base de datos de imágenes de entrenamiento para entrenar los descriptores planteados.
3. Evaluar el rendimiento del descriptor HOG y LBP utilizando métricas para resolver el problema de sobretamaño.
4. Realizar un análisis comparativo del descriptor de textura HOG y LBP.
5. Difundir los resultados en medios científicos nacionales o internacionales.

### **1.8.1. Principales contribuciones**

En este trabajo de tesis se hacen las siguientes contribuciones en el área de la clasificación de minerales:

- Un método para la clasificación de minerales en procesos industriales.
- Un análisis comparativo en términos de eficiencia de los descriptores estudiados de imágenes de mineral en 2D.

## **1.9. Estructura de la tesis**

En el capítulo II de este trabajo se ve la fundamentación teórica, donde se proporciona una introducción a los temas fundamentales para comprender este trabajo, como el reconocimiento de imágenes, el análisis de imágenes, los descriptores a trabajar y el clasificador implementado entre otros.

Una revisión sistemática de los trabajos existentes en el área de la clasificación de mineral en imágenes 2D obtenidas por cámaras y procesadas, es abordada en el capítulo III.

En el capítulo IV se describe el procedimiento paso a paso empleado para el desarrollo de la tesis. Como lo es la adquisición de la base de datos, el procesamiento de las imágenes, los parámetros de los descriptores y del clasificador ocupados en el proceso mostrando.

Finalmente, en el capítulo VI la discusión de los resultados y las conclusiones generadas son presentadas. Además, se encuentran líneas para posibles trabajos futuros que puedan continuar esta línea de investigación.

# Capítulo II

## Marco Teórico

### 2.1. Introducción

La siguiente sección muestra el sustento teórico presentado en la investigación, la cual va desde lo general a lo particular. Primero se da a conocer las disciplinas científicas que hacen parte de la investigación, esencialmente a los conceptos teóricos que la conforman. Seguido se explican los algoritmos para la extracción de características, el algoritmo para la clasificación lineal, y las métricas implementadas para la validación de la investigación.

### 2.2. Reconocimiento de Imágenes.

Algunas de las aplicaciones de análisis de imágenes es el reconocimiento del tamaño de rocas en forma automática. Porque la automatización permite establecer formas de control precisas y objetivas, a diferencia, de los sistemas manuales que se encuentran afectados al agotamiento y la rutina por parte del operador causando un control deficiente o inconsistente.

En general, numerosas aplicaciones se han implementado con la ayuda del análisis de imágenes, donde se han destacan distintos problemas como la detección de fallas en circuitos impresos, alimentos, láminas de silicio y granulometría entre otros. Sin embargo, hay que notar que cada análisis se encuentra directamente asociado al tipo de aplicación en el cual se desea emplear. Difícilmente un método diseñado para el reconocimiento de patrones de caras, por ejemplo, se

podrá aplicar a uno de circuitos impresos [17].

Sin embargo, el proceso de análisis de imágenes incluye un conjunto genérico de etapas bien definidas y que se adaptarán dependiendo del tipo de problema a resolver. Las etapas son mostradas en la Figura 2.1 al igual explicadas a contención [18]:

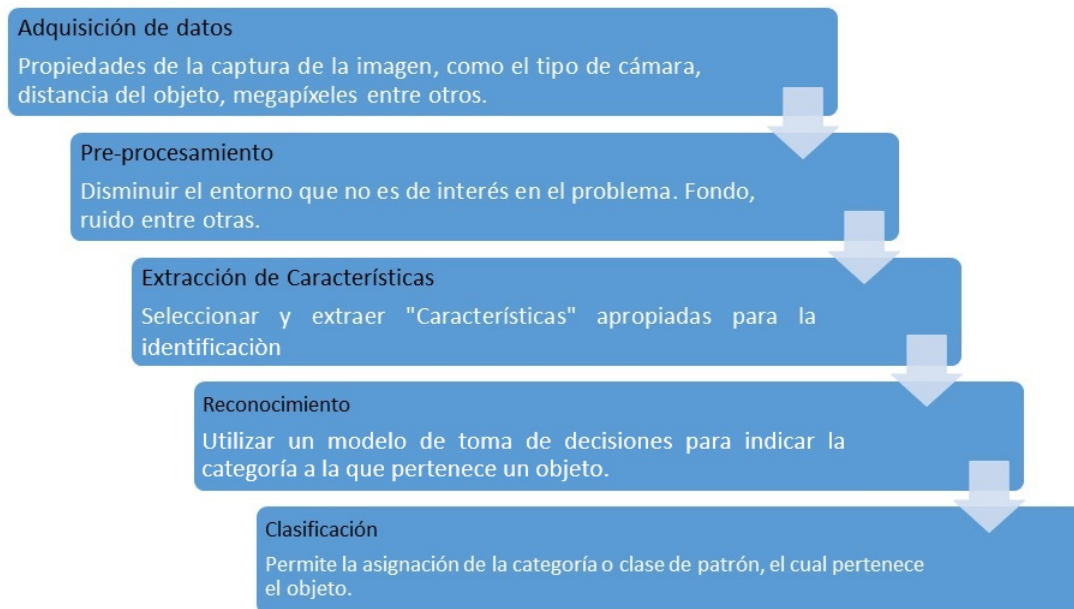


FIGURA 2.1: Análisis representativo del procesamiento de la Imagen.

1. **Adquisición de los Datos:** Para implementar la etapa de adquisición de los datos que describen el patrón a ser clasificado, se debe considerar la naturaleza del patrón. Por ejemplo, si se desea desarrollar un sistema para reconocer caracteres, esta etapa se podría realizar con la ayuda de una cámara digital o utilizando un escáner.
2. **Pre-procesamiento:** Después que se cuenta con los datos relevantes que describen cada uno de los patrones, es conveniente realizar una etapa de pre-procesamiento, el cual consiste en eliminar todos aquellos agentes externos que no vayan a ser analizados. Realizar el pre-procesamiento de los datos tiene varias ventajas siendo la principal de la reducción de la generalidad de los datos mejorando substancialmente la ejecución del sistema.
3. **Extracción de Características:** Los problemas en el reconocimiento de patrones es encontrar una manera óptima de representar la información original que describe a cada uno de éstos. Dicho problema es conocido como extracción de características el cual trata de reducir la cantidad de información (reducción de dimensionalidad) que representa a cada uno de los patrones, obteniendo de esta forma un agente de características que represente de la mejor manera posible al patrón original.

4. **Reconocimiento:** Cuando se ha realizado la etapa de pre-procesamiento y extracción de características se procede a realizar el reconocimiento. En general, esta etapa de reconocimiento consiste en obtener un patrón reconocido como respuesta a un conjunto de estímulos dados como entrada (imágenes).
5. **Clasificación:** Es el proceso, el cual se basa en las características extraídas (vector de características), permite asignar una categoría o clase a un patrón.

Por ejemplo si se tienen dos clases “+” y “o”, y el vector de características contiene la desviación estándar y la media. Si las clases son distantes, es posible mediante una línea recta poder distinguir entre ellas. Si un vector de característica “\*”, corresponde a un patrón desconocido, cae en la categoría de una región, se clasifica con esa clase.

## 2.3. Análisis de Imagen.

El enfoque de la propuesta de la tesis procede al análisis de dos descriptores para el análisis de imágenes. Donde cada imagen tiene una característica independiente dentro del proceso, con ello la variación de la medida del tamaño de la roca puede generar un cambio en el resultado final. Es así como es necesario analizar estas características de cada descriptor y proponer la mejor solución que permita mejorar el proceso de detección de tamaño de la roca.

### 2.3.1. Tipos de características a extraer de una Imagen

Uno de los pasos más importantes en la estructura de un seguidor de objetos en los análisis de imágenes, es la decisión de qué atributos son los más adecuados en el contexto del problema a solucionar. Algunas de las posibles opciones son:

**Color** Dentro del campo de color, existen varias representaciones de color como RGB, CMYK, entre otras. Cada uno tiene sus ventajas y desventajas, por lo que si se elige el color como el atributo de seguimiento de un objeto, se tendrán que tener en cuenta las dos propiedades físicas generales de este mismo: con qué tipo de luz se ilumina el objeto y las propiedades reflectantes de la superficie del objeto. Al utilizar este tipo de descriptor es muy importante tener en cuenta que si las condiciones de iluminación cambian, el espectro de color a seguir también cambiará [19].

**Esquinas** Este enfoque es uno de los más utilizados y existen varias opciones a considerar, entre las más importantes se encuentran Canny edge detector [20] y Harris [21]. Cada uno tiene diferentes ventajas y desventajas, sin embargo concuerdan que son sensibles a las imágenes con ruido, pero son menos sensibles a las variaciones de luz respecto al color debido a que los límites del objeto generan un cambio notable con el cálculo del gradiente o las intensidades de la imagen.

**Flujo óptico** Este enfoque consiste en calcular la diferencia de intensidad de un *template* en una imagen a través del tiempo para calcular el vector de desplazamiento de los píxeles que la conforman. Para lograr esto existen varios enfoques ya expuestos anteriormente: el **aditivo** [22] propuesto por Lukas-Kanade; el **aditivo inverso** [23] por Hager-Belhumeur y el **composicional** [24], de Shum y Szeliski, que incorporó mejoras en el rendimiento del algoritmo. Adicionalmente el **composicional inverso** [25] presentado por Baker-Matthews ofrece una mejora eficiente similar a la propuesta por Hage-Belhumeur. [23]. Este enfoque es muy utilizado en el seguimiento de objetos y segmentación basada en movimiento.

**Textura** Este enfoque a diferencia del color, requiere un paso extra, el cual genera los descriptores con base en la consistencia y rugosidad de la superficie del objeto. Dentro de estos métodos existe el *local binary patterns* (LBP) [8], *Histogram of Oriented Gradients* (HOG) [26], wavelets [27], entre otros. Este método también presenta una robustez a los cambios de luz a comparación del enfoque de color.

Como se puede observar, cada método propuesto funciona en diferentes condiciones. Es necesario considerar cuál descriptor se usará, ya que es parte fundamental del desempeño de la solución desarrollada.

La selección del atributo posee un gran impacto en la solución desarrollada. Muchas veces, no se cuenta con la experiencia o conocimientos suficientes para seleccionar el atributo óptimo para un problema determinado, por este motivo, se realizan investigaciones para atacar el problema con diversos métodos de selección automática de atributos, como se puede ver en [28].

### 2.3.2. Descriptores

Son vectores que caracterizan a un objeto multimedia, toman valores en los reales, en los que cada dimensión recibe un significado según el tipo de característica que se esté midiendo. Permiten

enfrentar el problema de la búsqueda por similitud en un espacio métrico empleando funciones de distancia que representan qué tan parecido es un objeto a otro.

Por otra parte, es conveniente realizar una distinción entre descriptores globales y locales, ambos presentes en el sistema [29]:

**Descriptores Globales** : Los descriptores globales hacen uso de la imagen en su totalidad para su descripción, permitiendo generalizar el objeto que se desea reconocer en un simple vector. Debido a esto, su uso en técnicas de clasificación es sencillo y a su vez el uso de recursos computacionales requerido es menor que en técnicas de clasificación con descriptores locales. Sin embargo, los descriptores globales al utilizar la imagen como un todo para la clasificación, son sensibles a las oclusiones y/o a fondos no homogéneos.

**Descriptores Locales** : Los descriptores locales se basan en la implementación de la extracción de características de múltiples regiones de una imagen. En términos generales el proceso que se realiza es el análisis de vecindad de los píxeles de una región definida, con lo que se obtiene un conjunto de vectores que describen cada punto de interés de la imagen. Al describir cada punto de interés de la imagen, permiten tener un clasificador robusto a oclusiones y a diversos ruidos que pudieran estar presentes.

A continuación se describen los descriptores utilizados en el proyecto:

### 2.3.2.1. Patrón Local Binario (LBP)

Este descriptor introducido por T Ojala, basado en texturas. El descriptor de patrones locales binarios (LBP) se desarrolló como medida complementaria de contraste, luego fue el primer método implementado para el reconocimiento de texturas en la detección de objetos en tiempo real y reconocimiento facial, logrando buenos resultados.

LBP es una medida, la cual es invariante a la escala de grises, derivado de una definición general de la textura en una vecindad local [30]. La propuesta de Harwood y Ojala [31] inicialmente fue la de comparar el pixel central con sus respectivos vecinos, donde el pixel central es utilizado como el umbral con respecto a su vecindad, al realizar la comparación entre el pixel central y sus vecinos se le asignara un valor de “1” en caso que este sea mayor o igual a ellos, o un cero “0” en caso contrario, donde a cada peso del umbral se le asignara  $2^n$  en donde  $n$  depende de la posición

del vecino con respecto al pixel central. Al finalizar se realizara una suma de los distintos pesos obteniendo la representación LBP del pixel, mostrado en la Figura 2.2.

Muestras		
10	22	25
23	20	21
9	17	20

(a)

Umbral		
0	1	1
1		1
0	0	1

(b)

Pesos		
1	2	4
128		8
64	32	16

(c)

FIGURA 2.2: Operador LBP donde (a) es un ejemplo para una muestra de una imagen de 3x3 pixel, (b) es su equivalente binaria del LBP y (c) los pesos asignados para cada pixel.

La ecuación II.1 muestra un ejemplo del valor obtenido al utilizar el operador LBP sobre los valores de intensidad representados en la Figura 2.2 con matrices de 3x3 y 8 vecinos.

$$LBP = (0 \times 1) + (1 \times 2) + (1 \times 4) + (1 \times 8) + (1 \times 16) + (1 \times 0) + (1 \times 0) + (1 \times 128) = 158 \quad (\text{II.1})$$

Posteriormente al operador LBP se le agregaron dos parámetros los cuales permiten la modificación del tamaño del área de interés y los vecinos para el píxel de interés por Ojala [32]. Los parámetros son definidos como  $R$  y  $P$ , donde  $R$  corresponde al radio o (Línea recta que une el centro de un círculo con cualquier punto del borde de la circunferencia) y  $P$  es el número de vecinos, definidos por el radio, que serán utilizados para el cálculo del LBP, ver Figura 2.3.

A partir de una circunferencia simétrica se construye la distancia (Radio) y distribución de los vecinos (Puntos), donde el intervalo debe ser para  $R > 0$  y  $P > 1$ , al modificar estos valores se obtendrán distintas medidas de textura en las regiones de interés. La Figura 2.3 muestra un ejemplo para una distribución con igual número de vecinos para distintos radios. Otro ejemplo de ajuste de parámetros se muestra en la Figura 2.4, donde el radio  $R$  tiene un valor constante igual a uno y la cantidad de vecinos  $P$  es cambiada para fines ilustrativos.

La representación de los vecinos repartidos circularmente, con respecto al pixel central se encuentra expresada matemáticamente en las dos ecuaciones:

$$x = R \sin \frac{2p\pi}{P} \quad (\text{II.2})$$

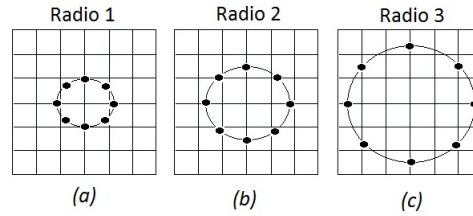


FIGURA 2.3: Distribución de los vecinos para un incremento del radio. (a)  $R = 1, P = 8$ , (b)  $R = 2, P = 8$ , (c)  $R = 3, P = 8$ .

$$y = R \cos \frac{2p\pi}{P} \quad (\text{II.3})$$

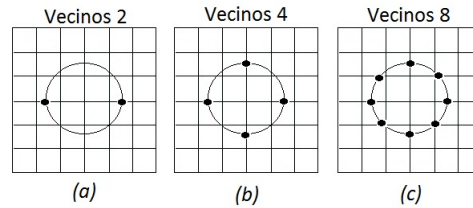


FIGURA 2.4: Distribución de una cantidad de puntos (vecinos) con un incremento de radios para LBP. (a)  $R=1, P=2$ , (b)  $R=1, P=4$ , (c)  $R=1, P=8$

Ya definido los parámetros del radio y la cantidad de vecinos (Puntos) el operador LBP es representado matemáticamente para  $R>0$  y  $P>1$  como ecuación II.4:

$$LBP_{p,r}(X_c, Y_c) = \sum_{P=0}^{P-1} 2^P S(g_p - g_c) \quad (\text{II.4})$$

Donde  $x_c, y_c$  corresponden a la posición del píxel de interés de la imagen  $g_p$ , es el valor de píxel central,  $g_c$  el valor los vecinos circulares del píxel central, y  $2^P$  el peso asignado a cada operación entre el píxel central y el vecino, finalmente  $S(g_p - g_c)$  se asignan valores de 1 y 0 según la ecuación II.5.

$$S(g_p - g_c) = \{1, g_p - g_c \geq 0, g_p - g_c < 0\} \quad (\text{II.5})$$

Una vez obtenido el valor del LBP para los píxeles de la imagen, se obtiene el histograma de la

imagen, dando como resultado un vector de  $N$  elementos.

Donde se calcula la frecuencia de los valores para obtener el Histograma dependiendo del tamaño de los *Bins* (Regiones Iguales) que divide el espacio de los posibles valores  $[0, 255]$  por ser a escalas de grises, como ejemplo se puede ver la Figura 2.5 , en el cual se divide el histograma en 13 *Bins*, de la Figura 2.6 la cual representa los valores LBP con 8 vecinos y radio 1.

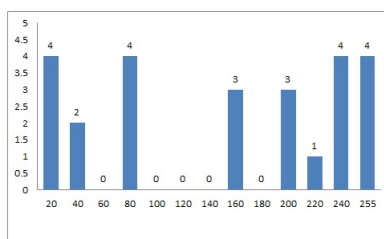


FIGURA 2.5: Histograma de ejemplo utilizando 13 Bins

	1	2	3	4	5	6	7	
1	0	0	0	0	0	0	0	1
2	0	40	192	248	24	160	0	2
3	0	6	207	143	159	3	0	3
4	0	6	191	255	239	3	0	4
5	0	62	63	255	231	195	0	5
6	0	62	62	255	227	227	0	6
7	0	0	0	0	0	0	0	7

FIGURA 2.6: Representación visual de los valores LBP con 8 vecinos y radio 1

La propuesta del operador LBP descrita ha sido recopilada por diferentes autores, dentro de estas denominadas extensiones compiladas por Mäenpää [31] se encuentran: Modelo estocástico y estructural de la textura, clasificación no paramétrico, invariancia de rotación, contraste y textura de los patrones, multi-resolución LBP y oponente de color LBP (OCLBP) [33].

**Invariancia a la Rotación:** La primera mejora que se propuso, de gran importancia para el campo de la descripción de textura, fue la invariancia a la rotación. Atendiendo a la definición de *Local Binary Pattern*, es fácil darse cuenta que el método no es invariante a la rotación. En la Figura 2.7 podemos ver un ejemplo de un patrón una vez aplicado el umbral a la diferencia de valor de gris, y el mismo rotado 90 grados, viendo que el resultado final es diferente.

Para corregir esta falta de robustez, se propuso un método basado en asignar a cada píxel el menor valor resultante de LBP de entre todas las posibles rotaciones del patrón. En la Figura

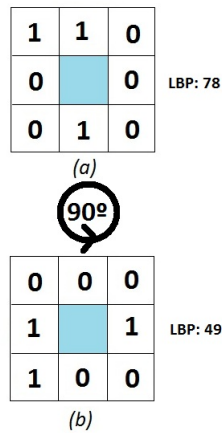


FIGURA 2.7: Ejemplo de dos patrones iguales pero con diferente orientación.

2.8 se pueden ver todas las posibles combinaciones de un patrón concreto y el valor asignado, correspondiente a la rotación 7 [34].

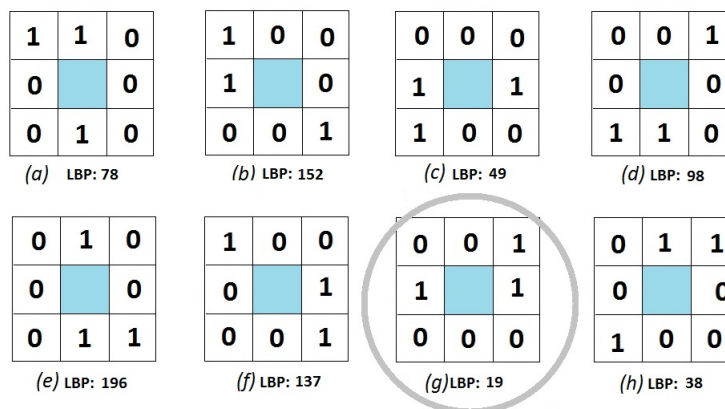


FIGURA 2.8: Ejemplo de todas las posibles rotaciones de un vector y su valor de LBP. En el método invariante a la rotación, se selecciona el valor mínimo para representar al píxel central [34].

La ecuación II.6 permite obtener una descripción invariante a la rotación:

$$LBP_{p,r}^{ri} = \min\{ROR(LBP_{p,r}, i), i = 0, \dots, P - 1\} \quad (\text{II.6})$$

Donde  $ROR(x, i)$  es un desplazamiento hacia la derecha de  $i$  píxeles para el patrón  $x$ .

**Uniformidad:** T. Ojala y otros [8] en el año 2007 desarrollaron una nueva propuesta para el cálculo de los descriptores LBP y que considera que la mayoría de la información relevante de la textura podía ser descrita mediante un patrón denominado uniforme [35]. Se puede definir

patrón uniforme como aquel patrón que tiene como mucho dos saltos entre 0 y 1 o entre 1 y 0 en su cadena, asumiéndose que es un patrón circular, por lo que es necesario comprobar también si hay transición entre el último elemento y el primero. En la Figura 2.9 podemos observar los distintos patrones uniformes que existen y algunos ejemplos de aquellos que no lo son.

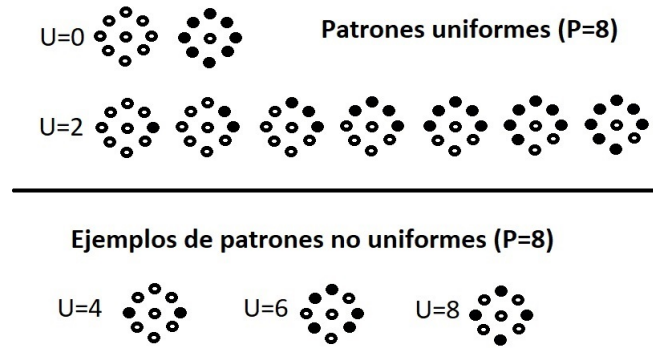


FIGURA 2.9: Patrones uniformes y no uniformes en LBP.

En la Figura 2.10 podemos ver las primitivas que representan los valores de algunos de los patrones uniformes. Teniendo en cuenta únicamente estos patrones, cualquier píxel de la imagen se puede representar mediante  $P + 2$  valores diferentes, asignando desde 0 hasta  $P$  el valor que corresponde con el número de píxeles cuyo valor es 1, y el valor  $P + 1$  a cualquier patrón no uniforme.



FIGURA 2.10: Representación de texturas definidas LBP.

En la ecuación II.7, se define el cálculo del LBP uniforme.

$$LBP_{p,r}^{riu2} = \sum_{P=0}^{P-1} S(g_p - g_c) \text{ Si } U(LBP_{p,r}) \leq 2, P + 1, \quad (\text{II.7})$$

Donde  $U(LBP)$  se define como indica la ecuación II.8.

$$U(LBP) = |S(g_{p-1}, g_c) - S(g_p, g_c)| \sum_{P=1}^{P-1} S(g_p - g_c) - S(g_{p-1} - g_c) \quad (\text{II.8})$$

Si el valor de  $U$  para un determinado patrón del LBP es menor o igual que 2, se designa como patrón uniforme. Los patrones que tienen un valor de  $U$  mayor que 2 son catalogados como no uniformes. El patrón  $L1 = 00001000$  es uniforme porque  $U(L1) = 2$ , mientras que el patrón  $L2 = 01100100$  es no uniforme porque  $U = 4$ .

**Histograma ULBP por Bloques:** Los histogramas se usan como descriptores eficientes de la imagen, especialmente cuando son transformados a su código ULBP pero en ocasiones determinadas, su invariante global ante la traslación produce efectos indeseados en el comportamiento del identificador, por lo cual, se recomienda el uso de descriptores por bloques dentro de la misma imagen [36].

Esto se ejemplifica en la Figura 2.11, si la aplicación deseada requiriera que se detectara, por ejemplo una persona, sin importar si la misma se encuentra en varias imágenes desordenadas, entonces el método estudiado funcionaría sin problema alguno, ambas imágenes tiene los histogramas con un mayor grado de similitud.

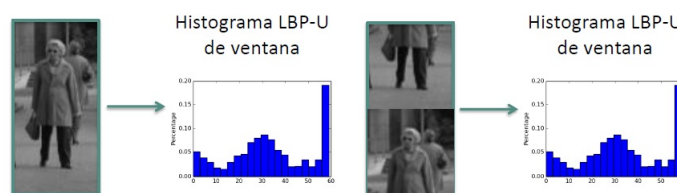


FIGURA 2.11: Representación de un mismo histograma LBP para imágenes distintas [37].

Pero como se muestra en la Figura 2.12 siguiente, si no se aplican los histogramas por bloques, el programa detectaría erróneamente la presencia de 5 personas cuando se trata únicamente de 3 una sobre la otra.

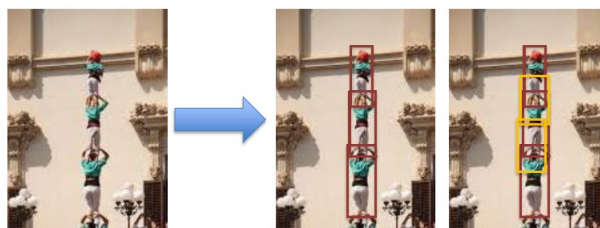


FIGURA 2.12: Ejemplo de aplicación errónea de histogramas sin bloques – Tomado del curso “Detección de objetos – Universidad Autónoma de Barcelona” disponible en Coursera.org.

Para solucionar este problema, se ha propuesto sub-dividir la imagen  $I$  en  $N$  secciones menores y analizar cada uno de los histogramas de cada sección por separado mostrada en la Figura 2.13. En la práctica se suele usar mallados de  $16 \times 16$  píxeles.

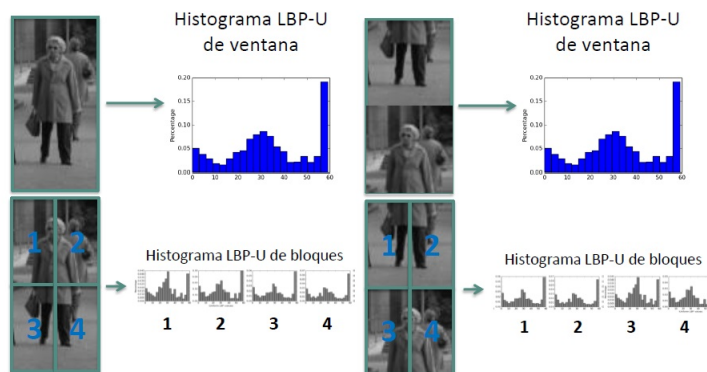


FIGURA 2.13: División de la imagen I de N bloques.

Finalmente, se debe concatenar los diferentes histogramas calculados en el orden en que fueron sub-divididas las  $N$  secciones, obteniendo un vector de  $N * 59$  donde cada fila representa el histograma ULBP de una sección  $N$ , convirtiéndose el vector completo en un identificador adecuado mostrado en la Figura 2.14.

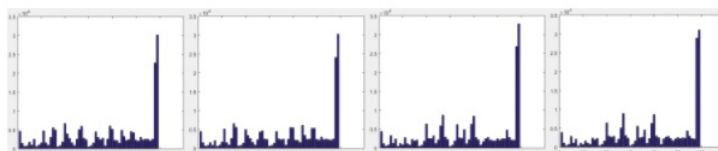


FIGURA 2.14: Histogramas individuales de los N bloques.

Una práctica usual es intercalar rejillas de mallado para obtener conjuntos de sub-secciones intercaladas, lo cual agrega cierta redundancia y permite ganar robustez en el programa mostrada en la Figura 2.15.

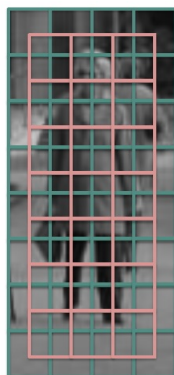


FIGURA 2.15: Intercalar rejillas de mallado de N bloques.

**Normalización de Histogramas** : Para aplicaciones de visión por computador es en ocasiones de gran utilidad realizar la normalización del histograma. Dada una imagen  $I$  de  $I_x, I_y$  dimensiones y un vector  $h$  que describe el histograma ULBP de una sección  $N$  mostrado en la Figura 2.16 [36].

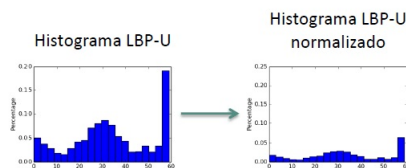


FIGURA 2.16: Histogramas Normalizado de  $N$  bloques.

Al normalizar, la suma de todos los componentes de cada histograma debe ser uno y en consecuencia, la suma de los componentes del vector descriptor, compuesto por los histogramas, será igual a  $N$  (Ecuaciones II.9, II.10, II.11 y II.12).

$$|x_i| = \frac{x_i}{I_x * I_y} \quad (\text{II.9})$$

$$|VH| = (h_1(|X_1|; \dots |X_n|); \dots; h_N(|X_1|; \dots |X_n|)) \quad (\text{II.10})$$

$$\sum_{i=1}^N |X_i| = 1 \quad (\text{II.11})$$

$$\sum_{i=1}^N |h_i| = N \quad (\text{II.12})$$

### 2.3.2.2. Histograma de Gradiente Orientados (HOG)

Es un descriptor de características que fue presentado por *Navneet Dalal* y *Bill Triggs* en la conferencia *Computer Vision and Pattern Recognition (CVPR)* en el 2005, el descriptor es utilizado para visión artificial por computadora y en el procesamiento de imágenes [38].

HOG es un descriptor de características usado principalmente para la detección de objetos en secuencias de imágenes [39, 40]. Este consiste en dividir una imagen en sub-bloques que se distribuyen a lo largo y ancho de la misma imagen con un cierto grado de solapamiento entre ellos,

a partir de ahí se procede a dividir en celdas para luego calcular las magnitudes y orientaciones de los gradientes por cada uno de los pixel, calculando el histograma de sus gradientes y orientaciones sobre cada uno de estos bloques, y finalmente son almacenados en vectores. Este método tiene como principal ventaja su mayor confiabilidad y robustez ante cambios de iluminación, contorno de la imagen o distintos fondos y escalas [41].

Como se había mencionado anteriormente, a una determinada imagen  $I(x,y)$  se la divide en un número de sub-imágenes de igual tamaño para en lo posterior agruparlas en bloques con la mismas cantidad de celdas, el solapamiento que debe existir entre dichos bloques es tal que, al tener un avance del bloque horizontalmente este elimina la columna de celdas del lado izquierdo e ingresa una nueva columna de la derecha que a su vez, lo realiza verticalmente, eliminando la fila de la parte superior y agregando las celdas de abajo, logrando que los gradientes no se calculan uniformemente sobre una malla densa lo que implicaría un coste en tiempo de computación demasiado grande ya que se tendría que calcular un HOG por cada celda, sino que gracias a la división en sub-bloques y celdas el coste computacional se agilizaría considerablemente [42].

Para el cálculo del HOG se deben distinguir tres etapas fundamentales, las cuales son: Cálculo del gradiente de la imagen, el diseño de los histogramas ponderados para cada sub-región, y la construcción de los histogramas normalizados por cada uno de los grupos de sub-regiones. Donde las sub-regiones se conocen como celdas, y el grupo de las celdas se conoce como bloques [43]. Describiremos los pasos más relevantes para la detección del descriptor HOG a continuación al igual son mostrados en la Figura 2.17.

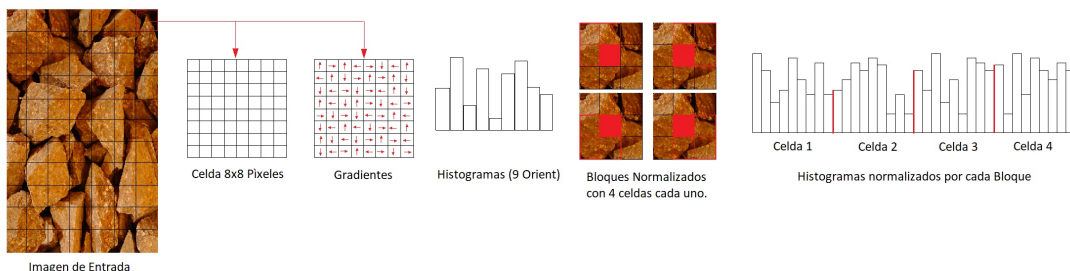


FIGURA 2.17: Proceso para el cálculo del descriptor HOG.

**Conversión a escala de Grises:** Primero se detecta una imagen  $I(x,y)$  a color RGB para su transformación a escala de grises  $I_g$ , donde esta transformación puede ser encontrada utilizando cualquiera de los métodos convencionales, como el del promedio de los canales, mostrado en la ecuación II.13, o el método de Bayer mostrado en la ecuación II.14, que pondera un peso mayor canal de verde, o el método de la transformación al espacio HSV,

mostrado en la ecuación II.15, donde la información de niveles de gris es obtenida del canal de brillo [44].

$$I_g = \frac{R + G + B}{3} \quad (\text{II.13})$$

$$I_g = \frac{R + 2G + B}{4} \quad (\text{II.14})$$

$$I_g = [0,299 - 0,587 - 0,114][R - G - B] \quad (\text{II.15})$$

Nótese que el método de Bayer y el de la transformación HSV producen resultados que son visualmente congruentes. El método del promedio es una implementación aritmética simple, pero sin mucho sentido físico. El método preferido en este trabajo es la transformación HSV de la ecuación II.15, ya que el canal de valor es una mejor representación de la luminancia [45].

**Cálculo del gradiente de la imagen:** Se calcula a través de operadores de derivadas discretas como: *Sobel*, *Prewitt*, *Roberts*, o el *Laplaciano*, [45]. Al aplicar estos operadores (*Kernel*) a las imágenes obtenemos los gradientes en las direcciones  $\nabla I_x$  e  $\nabla I_y$ . Utilizando las ecuaciones II.16 y II.17 se pueden determinar los componentes de magnitud,  $|\Delta I|$  de cada pareja de gradientes. También, se puede calcular la fase para los componentes  $x$  y del gradiente. La fase es dirección de la magnitud y este puede calcular utilizando las ecuaciones II.18 y II.19.

Para ilustrar el cálculo de los gradientes, en la Figura 2.18, se muestra una imagen de ejemplo que contiene dos valores de intensidades 0 y 255. Para calcular sus gradientes usamos las ecuaciones II.16 y II.17.

$$\nabla I_x = I(x + 1, y) - I(x - 1, y) = 255 \quad (\text{II.16})$$

$$\nabla I_y = I(x, y + 1) - I(x, y - 1) = 255 \quad (\text{II.17})$$

Adicionalmente, y una vez obtenidos los gradientes, es posible calcular la magnitud y orientación global de cada par de direcciones de gradiente. En la Figura 2.19, se ilustra los tipos de vectores y valores obtenidos al calcular los gradientes, magnitud y fase de una imagen.

0	0	255	255	255
0	0	255	255	255
0	0	0	255	255
0	0	0	0	0
0	0	0	0	0

Pixel Central

FIGURA 2.18: Ejemplo para el cálculo de los gradientes direccionales en un píxel dado (píxel central).

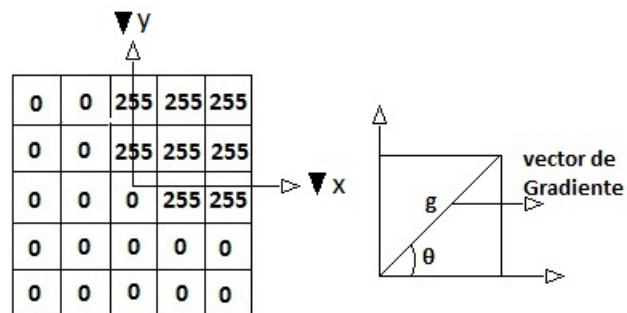


FIGURA 2.19: Cálculo de las orientaciones y las magnitudes en un píxel dado. Donde  $\nabla I_x$  e  $\nabla I_y$  son las direcciones respecto al eje  $X$  y  $Y$ ,  $g$  es la magnitud y  $\theta$  la orientación del píxel .

A partir del vector gradiente, se tiene:

$$\theta = \arctan\left(\frac{\nabla I_y}{\nabla I_x}\right) \quad (\text{II.18})$$

$$|\nabla I_x| = \sqrt{\nabla I_x^2 + \nabla I_y^2} \quad (\text{II.19})$$

**División de Celdas:** La celda es un sub-conjunto espacial local de la región de interés (ROI) la cual está formada por una cantidad de píxeles adyacentes [38]. Para representar una celda esta debe tener dos parámetros, uno de ellos es  $C_x$  el cual es el número de píxeles de la celda en el eje  $x$ , y  $C_y$  el número de píxeles de la celda en el eje  $y$ . El número total es representado en la ecuación II.20.

$$C = (C_x * C_y)(\text{píxeles}) \quad (\text{II.20})$$

Obtenidas  $C$  se divide cada una de las celdas en un tamaño fijo, por lo general este varía entre 6 y 8 píxeles tanto en ancho como en el alto, ver Figura 2.20, cabe recalcar que los valores pueden variar.

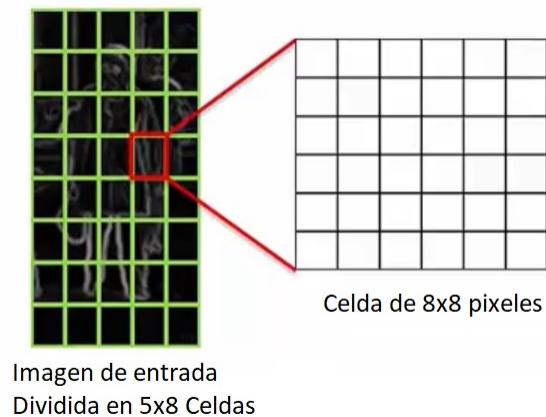


FIGURA 2.20: División de las celdas de tamaño fijos de píxeles.

Después la imagen es dividida en celdas con un tamaño fijo, tal y como se observa en la Figura 2.20, donde el número de celdas totales  $D$  en la imagen, es obtenida según la ecuación II.20, por ello se define  $C_x$  y  $C_y$  como la cantidad de celdas de la imagen en el eje  $x$  y  $y$  respectivamente. Para las ecuaciones  $n$  es el número de píxeles de la imagen en el eje  $x$  y  $m$  el número de píxeles del eje  $y$ .

$$D = D_x * D_y \quad (\text{II.21})$$

$$D_x = \frac{n}{C_x} \quad (\text{II.22})$$

$$D_y = \frac{n}{C_y} \quad (\text{II.23})$$

$$D = \frac{n}{C_x} * \frac{n}{C_y} \quad (\text{II.24})$$

A continuación se calcula el histograma de las orientaciones de los gradientes en cada una de las celdas, ver Figura 2.21.

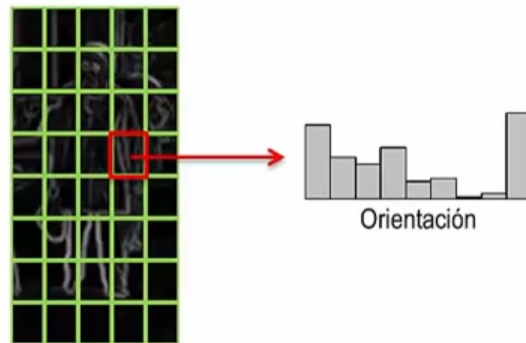


FIGURA 2.21: Histograma de orientaciones de gradientes en cada una de las celdas.

**Cálculo del Histograma de Orientación en una Celda:** Para ellos se procede como se ha mencionado anteriormente en dividir la celda en tamaños fijos. Se divide el rango de orientaciones en intervalos fijos, la primera consideración acepta al rango de orientaciones en sí mismo ya que puede considerar la orientación del gradiente con signo y con esto el rango de orientaciones estarían en el rango de  $[0^\circ, 360^\circ]$  o bien se lo puede considerar sin signo con lo que estaría en el rango de  $[0^\circ, 180^\circ]$ , tal y como se observa en la figura 2.22.

Es necesario convertir los valores de  $\theta$  a valores angulares que hagan parte del rango escogido. Estos nuevos valores obtenidos serán entonces los ángulos corregidos  $c$ , determinados según la ecuación II.25 y II.26:

Si el rango escogido es de  $[0^\circ, 180^\circ]$ :

$$\theta_c = \{\theta(x,y) + 180, \theta(x,y) < 0 \quad \theta(x,y), \theta(x,y) \geq 0\} \quad (\text{II.25})$$

Si el rango escogido es de  $[0^\circ, 360^\circ]$ :

$$\theta_c = \{\theta(x,y) + 360, \theta(x,y) < 0 \quad \theta(x,y), \theta(x,y) \geq 0\} \quad (\text{II.26})$$

Con esta orientación ( $0 - 180^\circ$ ) dos gradientes con la misma dirección pero sentidos inversos se consideran equivalentes y quedan asignados al mismo intervalo.

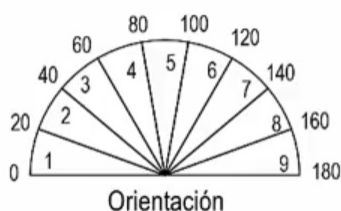


FIGURA 2.22: Diagrama de orientaciones e intervalos para los gradientes.

Como se observa figura 2.22, los intervalos de las orientaciones están en muestras de 9, lo cual permiten realizar los histogramas en dicha dimensión, como se puede observar en la Figura 2.23, la cual corresponde a 9 intervalos que suele ser un valor común, con un rango de  $20^\circ$  por intervalo.

Realizado esto, se asigna cada píxel de la celda a un intervalo en función de la orientación del gradiente, calculados previamente en los pasos anteriores, posteriormente se acumula la magnitud del gradiente de todos los píxeles asignados a un intervalo, obteniendo lo que se muestra en la Figura 2.23, aquí se observa que a cada intervalo le corresponde valores de la magnitud de los gradientes.

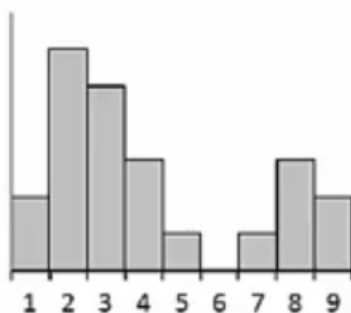


FIGURA 2.23: Acumulación de la magnitud del gradiente.

Cuando se tiene cambios de iluminación en las imágenes, la intensidad de los gradientes va a cambiar, por lo tanto las variaciones se reflejarán en los valores del histograma, es por ello que surge la necesidad de normalizar dichos valores con el objetivo de lograr que la magnitud

global del gradiente sea igual en todas las imágenes, entonces es necesario la normalización local adaptada por cada una de las zonas de interés de la imagen, de esta forma es introducido el concepto de bloque. El cual se verá en detalle en el próximo punto.

**Agrupación de las Celdas en Bloques:** Obtenidos los histogramas de cada celda de la imagen, se realizan agrupaciones de celdas en espacios  $b \times b$ . Estos nuevos espacios o regiones son llamados bloques. Para cada bloque  $k$  se concatenan los histogramas de las celdas  $h_{cij}$  que hacen parte de él, los cuales forman el vector  $\vec{v}$  que posteriormente será normalizado. El vector  $\vec{v}_k$  estará dado por la ecuación II.27:

$$\vec{v}_k = [[h_{11}][h_{12}]...[h_{ij}]...[h_{1b_x}][h_{21}]...[h_{2b_x}]...[h_{ij}]...[h_{b_y,1}][h_{b_y,2}]...[h_{b_x b_y}]] \quad (\text{II.27})$$

De la ecuación anterior los subíndices  $ij$  de  $h$  hacen representación de la posición de las celdas en el bloque,  $i$  el número de la fila y  $j$  el número de la columna que hacen parte del del bloque, donde:

$$i = 1, 2, 3, \dots, C_x \quad (\text{II.28})$$

$$j = 1, 2, 3, \dots, C_y \quad (\text{II.29})$$

Cada bloque agrupa un conjunto de celdas, por lo general, se utilizan dos celdas horizontales por dos celdas verticales, ver Figura 2.24.

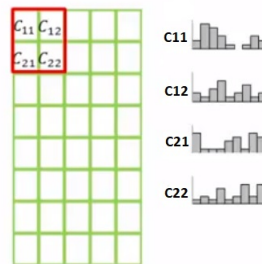


FIGURA 2.24: Histograma de cada una de las celdas de un bloque determinado..

Entonces para cada bloque le corresponde un histograma de cada una de las celdas, dichos histogramas se concatenan para obtener el vector con la representación del problema, ver Figura 2.25.

Para la normalización del vector  $\vec{v}$  es posible utilizar cualquiera de las normas convencionales tales como:  $L_2$ ,  $L_2$ -Hys,  $L_1$ , o  $L_1$ -sqrt [38, 46]. A continuación se definen cada una de ellas.

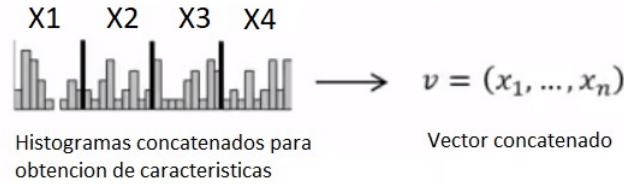


FIGURA 2.25: Histograma y Vector de características.

- Normalización con norma  $L2$ : Esta normalización llamada  $v_{L2}$  en la ecuación II.30 se calcula dividiendo cada uno de los componentes del vector  $v$  mostrada en la Figura 2.25 por la raíz cuadrada de la suma al cuadrado de la norma  $\|v\|_2$  más la constante  $\epsilon$ , donde la norma de un vector, es la raíz cuadra de la suma de cada uno de sus componentes al cuadrado mostrada en la ecuaciónII.31 y  $\epsilon$  es un valor pequeño, el cual se introduce con el fin de evitar di visiones por 0. En oportunidades donde la intensidad del bloque sea constante y la magnitud total del gradiente sea 0.

$$v_{L2} = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}} \quad (\text{II.30})$$

$$\|v_2\| = \sqrt{\sum x_i^2} \quad (\text{II.31})$$

- Normalización con norma  $L1$ : La norma  $L1$  corresponde igualmente a la división del vector  $v$  por la suma de la norma  $\|v\|_1$  mas la constante  $\epsilon$ . Mostrada en la ecuación II.32.

$$v_{L1} = \frac{v}{\|v\|_1 + \epsilon} \quad (\text{II.32})$$

- Normalización con norma  $L1-sqrt$ : Esta normalización es igual a la raíz cuadrada de la anterior, mostrada en la ecuación II.33.

$$v_{L1-sqrt} = \sqrt{\frac{v}{\|v\|_1 + \epsilon}} \quad (\text{II.33})$$

Donde  $\epsilon$  es una constante con un valor muy pequeño.

En la práctica, los bloques están definidos de manera que exista un solapamiento entre ellos, con el objetivo de encontrar el descriptor con mayor robustez ante variaciones en la forma del objeto; los bloques son colocados con una separación entre celdas tanto en horizontal como en vertical. Donde la representación del descriptor HOG es obtenida concatenando la

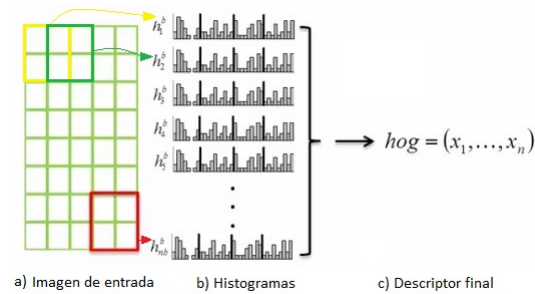


FIGURA 2.26: a) Se ilustra como se hace el recorrido del bloque por la imagen a procesar, teniendo un recorrido horizontal al igual vertical en ella. b) muestra cada uno de los histogramas calculados por cada uno de los bloques y c) el vector concatenado por cada uno de los histogramas de la imagen.

representación normalizada de los bloques solapados, tal y como se observa en la Figura 2.26.

De esta forma en la Figura 2.27(a) se observa que cada celda contribuye a la descripción de varios bloques, tantos como celdas se tenga en cada uno de los bloques, sin embargo pese a que dicha celda tendrá el mismo histograma en todos los bloques, cada uno de estos aplicará una normalización diferente, lo que hace que la contribución del histograma de la celda a los distintos bloques sea distinto como se muestra en la Figura 2.27(b).

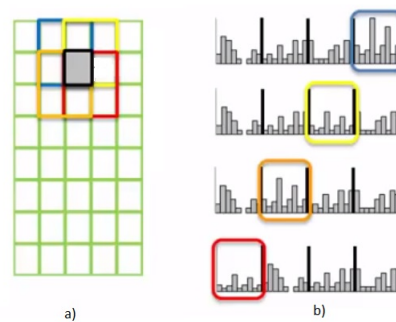


FIGURA 2.27: Contribución de las celdas a la descripción de varios bloques.

Para configurar el cálculo adecuado de un descriptor HOG es necesario evaluar el rendimiento de los siguientes parámetros antes de ser utilizados para la clasificación:

- Tamaño de celdas: Es la cantidad de píxeles el cual estará formado la celda.
- Signos del gradiente: Si para el cálculo del gradiente se ocuparan signos positivos o negativos.
- N° de intervalos del histograma de orientaciones: Es la división de los rangos de orientaciones.

- N° de celdas en cada bloque: Es la cantidad de celdas que componen un bloque.

Todos estos parámetros ayudan a determinar la dimensión  $n$  del descriptor final, este número final de dimensiones se puede calcular con la expresión dada en la ecuación II.34:

$$n = n^{\circ} \text{bloques} * \frac{\text{Tamaño de celdas}}{n^{\circ} \text{ de celdas en cada bloque}} * n^{\circ} \text{ intervalo histograma} \quad (\text{II.34})$$

De donde el número de bloques se encuentra aplicando lo siguiente ecuación II.35:

$$n^{\circ} \text{bloques} = n^{\circ} \text{celdas} - \frac{\text{Tamaño de celdas}}{n^{\circ} \text{ de celdas en cada bloque}} + 1 \quad (\text{II.35})$$

Realizado esto ya se conoce exactamente la dimensión del descriptor, es decir, el número de componentes que contendrá el mismo.

### 2.3.3. Detección del objeto

Una vez que se tiene decidido que atributos describen a nuestro objeto a seguir, se puede proceder a detectarlo. Para esto existen varias opciones dentro de las que se encuentran:

**Detectores de puntos** De la misma manera que los detectores de esquinas, los detectores de puntos son usados para encontrar características interesantes en una imagen. En algunos de estos detectores, existe la posibilidad de refinar la búsqueda de características interesantes a ciertos parámetros o umbrales, algunos ejemplos de estos detectores son SIFT [46], Harris [21], KLT [47] y Moravec [48]. Como en las demás secciones, cada uno tiene ventajas y desventajas dando como resultado tener que valorar que es más importante: la rapidez o la robustez.

**Background subtraction** Esta técnica genera un modelo del fondo de la escena, con el objetivo de calcular variaciones en esta misma, para detectar objetos en movimiento. Normalmente para detectar los objetos se utiliza un algoritmo de componentes conexas. Este enfoque ganó popularidad en el año 1997 con el trabajo de Wren y otros [49], con el que se calculaban pequeños cambios a través del tiempo.

**Segmentación** La finalidad de estos algoritmos es separar los objetos que sean parecidos en la imagen. Para lograr esto, se consideran dos factores: el primero es el criterio a seguir para separar los grupos de objetos similares y el segundo es encontrar el mejor método para

hacerlo. En esta categoría existen varios métodos dentro de los que se encuentran: Mean-shift Clustering [50], Graph-Cuts [51, 52] y Active Contours [53, 54].

**Aprendizaje supervisado** Esta técnica consiste en utilizar un algoritmo que es entrenado a partir del mapeo de ciertas características de la imagen con un valor pre-establecido, el cual puede ser continuo o una clase.

Es muy importante considerar qué características de la imagen se van a utilizar, ya que de ellas depende el desempeño y eficiencia del algoritmo. Algunos ejemplos de los algoritmos usados en este enfoque son: redes neuronales[55, 56], *adaptive boosting* [57], árboles de decisión [58], y máquinas de vectores de soporte [59].

Dentro del área del aprendizaje supervisado algunas de las técnicas más utilizadas son:

**Boosting** La técnica de *Boosting* consiste en generar iterativamente un clasificador preciso, con base en un gran número de clasificadores simples y con un porcentaje de acierto despreciable. Para generar estos clasificadores simples se pueden utilizar umbrales, donde en cada iteración se asignará un peso de votación a cada clasificador. Estos pesos serán asignados de acuerdo al mayor porcentaje de acierto que presente cada clasificador, esta técnica permitirá organizar los clasificadores del de mayor porcentaje de acierto hasta el menor.

**SVM** Las máquinas de soporte vectorial o *Support Vector Machine*, son clasificadores encargados de agrupar datos de manera separada de dos diferentes clases. Para realizar esta tarea, este clasificador se encarga de encontrar el mayor espacio marginal entre clases y los separa con un hiperplano. En caso de que el problema a resolver no sea lineal, es necesario aplicar un *kernel* dependiendo del tipo de problema para linealizarlo.

## 2.4. Aprendizaje Automático

También conocido como *Machine Learning* por sus siglas en inglés, es la rama de la inteligencia artificial, que mediante la ejecución de técnicas, permite dotar a las computadoras de la capacidad de “aprender”, es decir, crear programas con capacidad de generalizar comportamientos y identificar patrones a partir de la información provista en forma de ejemplos, de esta manera, al aprendizaje automático se lo conoce como un proceso de inducción del conocimiento, aquí se obtiene un enunciado general a partir de casos particulares, este método de aprendizaje se lo usa

comúnmente para el reconocimiento de rostros, clasificación de correos (*spam*), reconocimiento de voz, entre otros, en resumen, *Machine Learning* permite: extraer información valiosa que admite comprender una determinada situación o problema; procesar, clasificar, interpretar, analizar y evaluar información disponible; predecir escenarios y situaciones; generar aprendizaje para implementar mejoras y automatizar procesos [60].

El aprendizaje automático, posee técnicas que posibilitan la creación de los distintos algoritmos necesarios para el aprendizaje, por ello, *Machine Learning* se divide en dos áreas principales:

- **Aprendizaje supervisado:** Se basa en hacer predicciones a futuro fundamentándose en comportamientos o características que se observan en datos que están previamente almacenados (base de datos), permite buscar patrones en datos en la base de datos existente relacionando todo los campos con el campo objetivo [60].
- **Aprendizaje no supervisado:** Este hace uso de datos que no están etiquetados, el fin que se persigue con este método de aprendizaje, es explorar los datos para encontrar alguna estructura o manera de organizarlos [60].

Para el desarrollo de esta investigación se hace uso del primer método de aprendizaje, el aprendizaje supervisado, es por ello, que a continuación se hará énfasis, al igual en una de las técnicas mayormente usada dentro de este método, como lo son las Máquinas de Soporte Vectorial.

### **2.4.1. Aprendizaje Supervisado**

Tal y como se había descrito en párrafos anteriores, el aprendizaje supervisado hace uso de un algoritmo que origina una función estableciendo una correspondencia entre las entradas y salidas deseadas del sistema, el uso de este algoritmo se da principalmente en problemas relacionados con la clasificación, aquí el sistema de aprendizaje intenta etiquetar (clasificar) una serie de vectores por medio de varias categorías que se conocen comúnmente como clases.

Los pasos que se suelen seguir para la resolución de un problema mediante el aprendizaje supervisado, se describen a continuación [61]:

1. Determinar los objetos de entrenamiento y fijar el tipo de dato a emplear.

2. Adquirir un conjunto de datos de entrenamiento, ya sea de forma manual, mediante un sensor, ejemplos reales, entre otros.
3. Establecer el formato de los datos de entrada, es decir, partiendo de los ejemplos de entrenamiento del paso anterior, encontrar las características a estudiar para el desarrollo de una función que acople dichas características al entorno de programación.
4. Decidir la función que resolverá el problemas: elegir si se emplean arboles de decisión, redes neuronales, máquinas de soporte vectorial, etc.

Para el desarrollo de la presente tesis se seleccionó el clasificador de Máquina de Soporte Vectorial por ser una de las técnicas más utilizadas en este tipo de aplicaciones (análisis granulométrico). Además, se podrán comparar los resultados con otros tipos de publicaciones para poder estudiar su rendimiento bajo las mismas condiciones.

#### 2.4.1.1. Máquina de Soporte Vectores

Las Máquinas de Soporte Vectorial (SVM por su nombre en inglés *Support Vector Machine*) es una técnica o método de aprendizaje supervisado, que ha demostrado en los últimos años, tener un gran desempeño, superando a las máquinas de aprendizaje tradicional como las redes neuronales. Una SVM aprende de la superficie de dos clases distintas, en donde a partir de muestras de entrada se reconoce patrones que permiten resolver problemas de clasificación, en este caso se hará uso de este algoritmo para el reconocimiento de rocas desde un enfoque multiclase (Cuatro clases) [42].

Para la mejor comprensión de la técnica se analizará el problema de clasificación de dos clases, es decir, partiendo de un conjunto de muestras iniciales tanto positivas como negativas con  $f$  características cada una, se coloca cada una de las muestras como un punto en el espacio de  $d$  dimensiones, entonces el objetivo es encontrar una línea, conocida como hiperplano, que logre separar los puntos de ambas clases en dos grupos, tal y como se observa en la siguiente Figura 2.28 [42].

Los datos que se mapean con el clasificador SVM son realizados por medio de un *kernel*, a un espacio de características dimensionalmente mayor, por ejemplo: si los puntos de entrada están en  $R^2$  entonces son mapeados por la SVM a  $R^3$ , ahí encuentra un hiperplano que los separe y busca la máxima separación entre las clases, dicha distancia está marcada como margen en la Figura 2.29; cabe recalcar que un *kernel* es una función matemática que se emplea en las Máquinas de Soporte

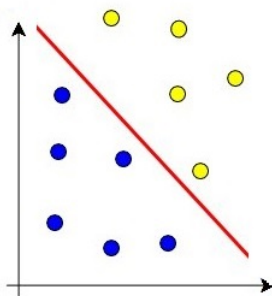


FIGURA 2.28: Hiperplano de separación de dos clases.

Vectorial que permite convertir problemas de clasificación no-lineal en el espacio dimensional original, a un problema de clasificación lineal en un espacio dimensional mayor, los *kernels* más comunes son: la función lineal, polinomial, RBF (*Radial Basis Function*), ERBF (*Exponential Radial Basis Function*), entre otros [61].

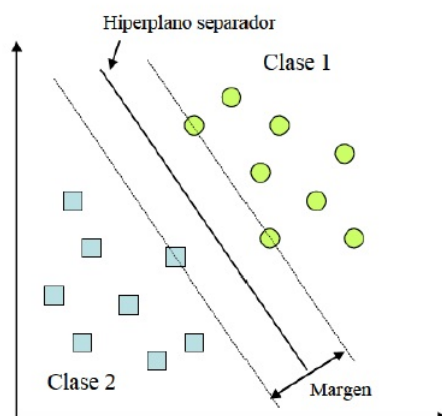


FIGURA 2.29: Separación de datos mediante SVM.

El caso más general es resolver problemas cuyos datos de entrada son linealmente separables. SVM es un clasificador lineal por lo que debe adaptarse cuando se utilizan datos que no son linealmente separables.

Para resolver los diferentes problemas de clasificación con SVM existen diferentes tipos de estrategias:

**SVM lineal con Margen Máximo:** Las Máquinas de Soporte Vectorial conforman hiperplano que separan los datos de entrada en dos subgrupos que tienen etiqueta propia en el caso de ser linealmente separable, es decir, en medio de todos los posibles planos de separación de las

dos clases que se etiquetan como  $\{-1, +1\}$ , existe únicamente un hiperplano de separación óptimo, de tal forma que suceda la maximización del margen y en donde se apoyan puntos denominados vectores de soporte, tal y como se observa en la Figura 2.30 [60].

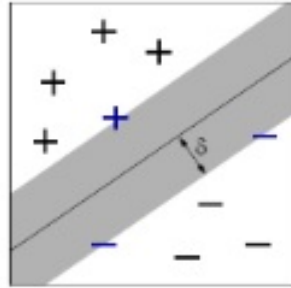


FIGURA 2.30: SVM linealmente separable.

Este caso solo se debería emplear cuando los datos son linealmente separables de tal manera que se cumpla la ecuación II.36:

$$h(x) = w^T x + b = 0 \quad (\text{II.36})$$

Donde  $x \in \mathbb{R}^d$ , siendo  $d$  la dimensión del espacio de entrada.

La resolución para dicho caso sería suponer que se tiene un conjunto de  $n$  datos separables linealmente mostrado en la ecuación II.37:

$$\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} \text{ donde } x_i \in \mathbb{R}^d \text{ e } y_i \in \{-1, 1\} \quad (\text{II.37})$$

Se cumplirá, según el lado en el que esté respecto del hiperplano mostrado en la ecuación II.38 II.39:

$$w^T x_i + b > 0, \text{ para } y_i = 1, \quad i = 1, \dots, n \quad (\text{II.38})$$

$$w^T x_i + b < 0, \text{ para } y_i = -1, \quad i = 1, \dots, n \quad (\text{II.39})$$

Las ecuaciones definen las dos clases presentes en este problema que al no estar mezcladas permiten hallar con sencillez a nivel matemático el hiperplano que las separa con margen máximo. Las expresiones antes halladas se las puede reducir a una sola ecuación II.40:

$$w^T x_i + b > 0, \text{ para } y_i = 1, \quad i = 1, \dots, n \quad (\text{II.40})$$

Para resolver el problema, se considera que los vectores soporte (con relleno en color negro en la Figura 2.30) es decir, los puntos más cercanos al hiperplano, cumplen Ecuación II.41, II.42:

$$h(x_i) = 1, \text{ para } y_i = 1 \quad (\text{II.41})$$

$$h(x_i) = -1, \text{ para } y_i = -1 \quad (\text{II.42})$$

En donde al ser una aproximación sencilla, los vectores soporte son fácilmente identificables a nivel gráfico. Se resalta que no puede haber datos del conjunto de aprendizaje dentro del margen, por lo que la ecuación  $y_i(w^T x_i + b) > 0$ , queda como se muestra en la ecuación II.43:

$$y_i(w^T x_i + b) \geq 1, \quad i = 1, \dots, n \quad (\text{II.43})$$

La distancia  $dist(h, x)$  de un punto al hiperplano es la ecuación II.44:

$$dist(h, x) = \frac{|h(x)|}{|w|} \quad (\text{II.44})$$

Como los puntos más próximos al hiperplano cumplen  $|h(x)| = 1$ , su distancia al hiperplano sería la ecuación II.45:

$$dist(h, x) = \frac{1}{|w|} \quad (\text{II.45})$$

Entonces para encontrar los valores de  $w$  y  $b$  hay que resolver un problema de optimización que consiste básicamente en maximizar la distancia  $dist(h, x)$  entre el hiperplano y el punto de entrenamiento más próximo, entonces:

Maximizar ecuación II.46:

$$\frac{1}{|w|} \quad (\text{II.46})$$

Sujeto a ecuación II.47:

$$y_i(w^T x_i + b) \geq 1, \quad i = 1, \dots, n \quad (\text{II.47})$$

Que es la condición que indica que ningún vector de entrenamiento debe quedar dentro del margen que separa a las dos clases. En la figura 2.31 se muestra todas estas situaciones con sus elementos característicos, es decir, las dos clases, el hiperplano y el margen máximo.

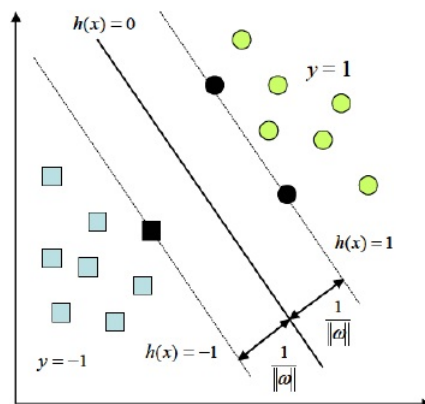


FIGURA 2.31: SVM con margen máximo (en negro los vectores de soporte).

Entonces se puede decir que la ecuación del hiperplano sólo depende de los vectores soporte, de ahí surge el nombre del algoritmo, máquinas de soporte vectorial.

**SVM para la clasificación no lineal:** Existen situaciones donde es posible que los datos de entrada no sean linealmente separables, como se puede apreciar en la figura 2.32.

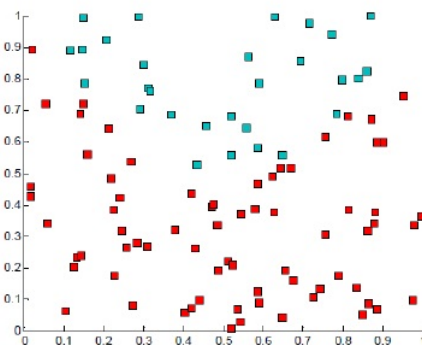


FIGURA 2.32: Conjunto de datos linealmente no separables.

A continuación se describen dos situaciones en la que se puede dar solución a este tipo de problemas:

- El primero requiere del uso de una función *Kernel* para la transformación de las variables de entrada ya que los datos pueden ser separables con margen máximo pero en un espacio de dimensionalidad mayor, ver figura 2.33.

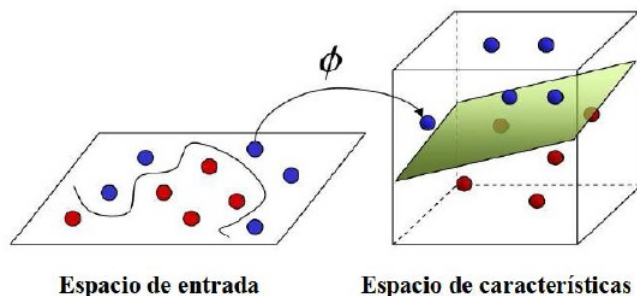


FIGURA 2.33: Transformación de datos de entrada a un espacio de dimensión mayor.

En la figura 2.34 se puede observar gráficamente como la función *Kernel* realizaría la separación y el traslado de los datos al espacio de características [60].

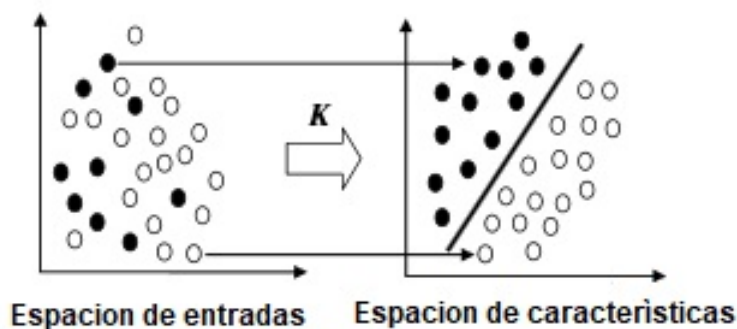


FIGURA 2.34: SVM no lineal inducida por una función Kernel.

- b) La segunda situación de las SVM se denomina como Margen Blando o “*Soft Margin*” por sus siglas en inglés, aquí no es viable encontrar una transformación de los datos, ya sea, en el espacio de entrada o en el espacio de características, es decir, no se puede separar linealmente los datos; trata aquellos casos donde los datos de entrada son erróneos, presentan ruido o solape entre las distintas clases de los datos de entrenamiento que comprometa el hiperplano clasificador [60].

### Máquinas de Soporte Vectorial para clasificación multiclase

Las máquinas de soporte vectorial (SVM) se usan comúnmente para problemas de tipo binario, pero puede darse el caso de que el problema a tratar implique datos de entrada de más de dos clases.

Existen dos métodos para resolver problemas de clasificación multiclase, entre ellos:

- a) Clasificación 1-v-r (*one-versus-rest*): En este método de clasificación el problema se considera una única clase positiva y el resto negativas, por ello se deben hallar los distintos hiperplanos según la cantidad de clases del sistema.
- b) Clasificación 1-v-1 (*one-versus-one*): En este método en cambio, para cada problema se toman dos clases de las  $K$  totales y se compara cada clase con cada una de las restantes, lo que supone realizar  $K(K - 1)/2$  clasificaciones; de estos dos métodos, el primer enfoque es el más habitual.

## 2.5. Evaluación del rendimiento de un clasificador

Todo algoritmo de clasificación está sujeto a la posibilidad de cometer errores en sus resultados. Por lo tanto, la evaluación del rendimiento de un clasificador es crucial para conseguir resultados aceptables una vez que el clasificador se ponga en producción.

Una primera aproximación de la evaluación del modelo consiste en aplicar el clasificador a los propios ejemplos proporcionados en el entrenamiento. Sin embargo, esta estimación genera resultados muy sesgados y conlleva a errores de sobre-ajuste [62]. Para obtener resultados más realistas, se utiliza un conjunto de datos en el que se proporcionan nuevos ejemplos que no se han incluido en el proceso de entrenamiento, para ello se divide el conjunto de datos en dos subconjuntos: uno con los datos para el entrenamiento y otro con los datos para la prueba.

Si el *dataset* es muy pequeño, el número de instancias disponibles para entrenar el modelo quedaría muy reducido, esto conllevaría a obtener peores resultados. Para solventar este inconveniente se utilizan técnicas de validación cruzada o *cross-validation* que proporcionan mayor objetividad a las pruebas. La técnica generalizada de *cross-validation* se conoce como *k-fold* la descripción de su funcionamiento y la de sus variantes se describen en la Tabla 2.1 [63].

El rendimiento obtenido de los  $k$  casos proporciona una estimación para saber cómo se va a comportar el modelo utilizando nuevos datos. Una de las estimaciones más importante en la inteligencia artificial es la precisión del modelo que se construye, ya que no nos servirá de nada una herramienta muy buena que obtuviese un mal rendimiento.

Una vez que se ha aplicado el modelo a varios *datasets* de prueba, se pueden obtener una serie de métricas que proporcionan mayor información acerca del rendimiento del clasificador.

Parámetros	Descripción
<i>2-Fold</i>	Se divide al <i>dataset</i> en dos subconjuntos de tamaños iguales $S_1$ y $S_2$ . Se utiliza el subconjunto $S_1$ para entrenar el modelo y luego se lo evalúa con el subconjunto $S_2$ . Luego se repite el proceso utilizando $S_2$ en el entrenamiento y $S_1$ en la prueba. Se elige el clasificador que tenga mejor precisión o menor error.
<i>k-fold</i>	Los $n$ datos se parten $k$ veces ( $k < n$ ) en 2 subconjuntos (normalmente $n/k$ para el <i>test</i> y $(n - n/k)$ para entrenamiento, excluyendo cada vez un subconjunto distinto.) El algoritmo se ejecuta $k$ veces y luego se elige el clasificador con mejor resultado.
<i>Leave one out</i>	Sigue el mismo proceso que en el <i>k - fold</i> pero para $k = n$ .

TABLA 2.1: Técnicas de validación cruzada.

El proceso de evaluación en la fase de desarrollo del clasificador es un proceso iterativo. Esto implica que por cada cambio realizado en el modelo, este último, se debe reevaluar con la finalidad de determinar el impacto sobre el rendimiento. Al finalizar la fase de evaluación es importante determinar qué clasificador ha logrado un nivel aceptable de rendimiento y a su vez representa una mejora considerable frente a otros clasificadores existentes.

## 2.6. Métricas

Para la etapa de experimentación se requieren maneras de medir el desempeño de los sistemas implementados específicamente la eficacia, ya que se ha dejado fuera el análisis de eficiencia, donde la primera se entiende como la capacidad de retornar la clase correcta para los objetos de consulta, lo cual es el tema que se trabaja en la investigación. Mientras que la segunda corresponde al tiempo que demora el proceso. Para efectos de este proyecto se considerarán métricas, cuyas definiciones fueron tomadas desde Baeza-Yates y Ribeiro-Neto [29].

Los valores que se han obtenido son con el apoyo de la matriz de confusión multiclase que se describirá a continuación, por la naturaleza de la investigación, la cual permite tener con claridad los resultados presentados.

		Condición Predichas	
		Condición Positiva	Condición Negativa
Condición Real	Condición Real Positiva	<b>Verdadero Positivo</b> <i>True positive (TP)</i>	<b>Falso Positivo</b> <i>False positive (FP)</i>
	Condición Real Negativa	<b>Falso Negativo</b> <i>False Negative (FN)</i>	<b>Verdadero Negativo</b> <i>True Negative (TN)</i>

TABLA 2.2: El gráfico muestra el conjunto de terminologías y derivaciones de una matriz de confusión.

### 2.6.1. Matriz de confusión Multiclase

En el campo del aprendizaje automático y específicamente el problema de la clasificación estadística, una matriz de confusión, también conocida como matriz de error, [64] es un diseño de tabla específico que permite la visualización del rendimiento de un algoritmo mostrada en la Tabla 2.2. Donde cada fila de la matriz representa las instancias en una clase real, mientras que cada columna representa las instancias en una clase predicha(o viceversa) [65]. El nombre proviene del hecho de que hace que sea fácil ver si el sistema está confundiendo las clases (es decir, etiquetar erróneamente unas como otras).

La Tabla de confusión tiene una terminología y característica fija, la cual describiremos y mostraremos en detalle en la Tabla 2.2.

- **TP:** Es el número de predicciones correctas de que una instancia es negativa.
- **FP:** Es el número de predicciones incorrectas de que una instancia es positiva.
- **FN:** Es el número de predicciones incorrectas de que una instancia es negativa.
- **TN:** Es el número de predicciones correctas de que una instancia es positiva.

Como la matriz de confusión hace relación a la precisión y esta no es una métrica confiable, con la tabla se llevan a nuevas cálculos de métricas que nos permite buscar una mejor confiabilidad en nuestro proceso, los cuales se detallaran a continuación:

**La Accuracy: (ACC)** es la proporción del número total de predicciones correctas. Se determina usando la ecuación II.48:

$$\mathbf{ACC} = \frac{TP + TN}{TP + FP + FN + TN} \quad (\text{II.48})$$

La tasa de recuerdo o de verdadero positivo (**TP**): Es la proporción de casos positivos que se identificaron correctamente, según se calculó utilizando la ecuación II.49:

$$\mathbf{TP} = \frac{TN}{FN + TN} \quad (\text{II.49})$$

La tasa de falsos positivos (**FP**): Es la proporción de casos negativos que se clasificaron incorrectamente como positivos, según se calcularon usando la ecuación II.50:

$$\mathbf{FP} = \frac{FP}{TP + FP} \quad (\text{II.50})$$

La tasa negativa verdadera (**TN**): se define como la proporción de casos negativos que se clasificaron correctamente, según se calculó usando la ecuación II.51:

$$\mathbf{TN} = \frac{TP}{TP + FP} \quad (\text{II.51})$$

La tasa de falsos negativos (**FN**): es la proporción de casos positivos que se clasificaron incorrectamente como negativos, según se calculó usando la ecuación II.52:

$$\mathbf{FN} = \frac{FN}{FN + TN} \quad (\text{II.52})$$

Finalmente, la precisión (**P**): Es la proporción de casos positivos predichos que fueron correctos, según se calculó usando la ecuación II.53:

$$\mathbf{P} = \frac{TP}{FP + TP} \quad (\text{II.53})$$

Para el caso de evaluaciones sobre sistemas multiclase, donde existen varias clases (minerales en este caso) dentro de las cuales se pueden clasificar los elementos disponibles, requieren de una tabla mas compleja como se muestra en la Tabla 2.3. Dicha Figura esta organizada por

	Condición Predicha				
Condición Real	$M_{11}$	$M_{12}$	$M_{13}$	...	$M_{1i}$
	$M_{21}$	$M_{22}$	$M_{23}$	...	$M_{2i}$
	$M_{31}$	$M_{32}$	$M_{33}$	...	$M_{3i}$
	...	...	...	...	...
	$M_{j1}$	$M_{j2}$	$M_{j3}$	...	$M_{ji}$

TABLA 2.3: El gráfico muestra un conjunto de observaciones distintos tipos de clases.

dos elementos principales: filas que representan los elementos de predicción correctos para la clasificación (Predicciones) y las columnas que reflejan los resultados obtenidos por el sistema (Actuales).

En las matrices de confusión multiclase al igual que en las matrices de confusión, se puede identificar el estado de los elementos clasificados mediante la realización de sumatorias, donde:

Los verdaderos positivos se definen según la ecuación: II.57

$$\sum_i M_{ii} \quad (\text{II.54})$$

Los falsos negativos para cada clase  $i$  se define como II.58

$$\sum_i M_{ji}, \text{ donde } j \neq i \quad (\text{II.55})$$

Los falsos positivos para cada clase  $i$  se define como II.56

$$\sum_i M_{ij}, \text{ donde } j \neq i \quad (\text{II.56})$$

Como se puede apreciar en las fórmulas anteriormente descritas, los valores que se pueden obtener tanto para falsos positivos y falsos negativos son por clases. Lo que significa que se podría aplicar las fórmulas definidas anteriormente para matrices de confusión simples, con el pequeño inconveniente que solo daría valores parciales por cada clase.

[!hbtpt]

		Predichos		
		Perros	Gatos	Conejos
Real	Perros	13	0	0
	Gatos	0	10	6
	Conejos	0	0	9

TABLA 2.4: El gráfico muestra un conjunto de observaciones de tres tipos de clases, las cuales son Perros, Gatos y Conejos. Los cuales se tiene una cantidad real por cada uno de ellos y las predicciones que se realizaron al momento de ser calculadas por el clasificador.

La Tabla 2.4 es un ejemplo de un sistema de clasificación multiclase entrenado para distinguir tres tipos de animales, entre ellos están perros, gatos y conejos, la matriz resumirá los resultados de probar el algoritmo de clasificación multiclase. Suponiendo una muestra de 38 animales: de los cuales 13 son perros, 16 gatos y 9 conejos.

En esta matriz de confusión de la Tabla 2.5, de los 16 gatos reales, el sistema predijo que seis de ellos son conejos. Mientras que las entre perros y conejos las reales y predichas fueron correspondidas en la cantidad deseada. Podemos ver en la matriz que el sistema en cuestión tiene problemas para predecir gatos, pero puede hacer bastante bien la distinción entre perros y conejos. Todas las predicciones correctas están ubicadas en la diagonal de la tabla, por lo que se es fácil inspeccionar visualmente la tabla en busca de errores de predicción, ya que estarán representadas por valores fuera de la diagonal.

Esta matriz nos permite un análisis más detallado y no solo la clasificación correcta (Precisión). Debido que la precisión no es una métrica confiable para el redimiendo real de un clasificador, ya que arrojará resultados engañosos si el conjunto de datos está desequilibrado (es decir, cuando el número de observaciones en diferentes clases varía mucho). Por ejemplo, si había 95 gatos y solo 5 perros en los datos, un clasificador en particular podría clasificar todas las observaciones como gatos. La precisión general sería del 95 %, pero con más detalle el clasificador tendría una tasa de reconocimiento (sensibilidad) del 100 % para la clase de gato pero una tasa de reconocimiento del 0 % para la clase de perro.

## 2.6.2. Curvas Roc

La metodología ROC fue desarrollada en el contexto de la detección de señales electrónicas en los inicios de la década de los 50. A mediados de los 60 se habían usado las curvas ROC en psicología

		Predichos	
		Gatos	No Gatos
Real	Gatos	10 Verdaderos Positivos	6 Falsos Positivos
	No Gatos	0 Falsos Negativos	38 Verdaderos Negativos

TABLA 2.5: El gráfico muestra el conjunto de observaciones obtenidas por los Gatos.

y psicofísica experimental [66]. Leo Lusted, radiólogo, fue el primero en usarlas en el proceso de toma de decisiones médicas, en 1967, y comenzó a aplicarlas en estudios con imágenes, en 1969 [67].

Las curvas ROC proporcionan un buen índice de la capacidad de una prueba diagnóstica para discriminar entre estados alternativos de salud cuando los resultados son medidos en escala ordinal, por intervalo o continua [68, 69]. Son útiles también para seleccionar umbrales de decisión (puntos de corte entre los resultados positivos y negativos de la una prueba) [70, 71]. La precisión de una prueba diagnóstica puede ser medida en términos de sensibilidad y especificidad [72].

Las curvas ROC son gráficas bidimensionales que trazan la tasa de verdaderos positivos versus la tasa de falsos positivos, es decir, representan la compensación entre los beneficios (TP) y los costos (FP), mostrado en la Figura 2.35

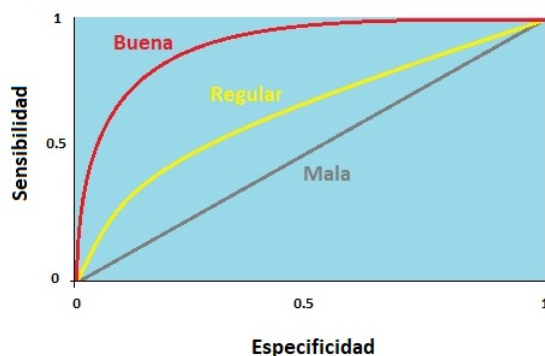


FIGURA 2.35: Ejemplo del comportamiento de la curva Roc.

**La sensibilidad:** Es la probabilidad de clasificar correctamente a un individuo cuyo estado real sea el definido como positivo respecto a la condición que estudia la prueba, razón por la que también es denominada fracción de verdaderos positivos (FVP). Se determina usando la ecuación II.57 con ayuda de la Tabla de matriz de confusión.

$$\text{Sensibilidad} = \frac{TP}{TP + FN} \quad (\text{II.57})$$

**La especificidad:** Es la probabilidad de clasificar correctamente a un individuo cuyo estado real sea el definido como negativo. Es igual al resultado de restar a uno la fracción de falsos positivos (FFP).

$$\text{Especificidad} = \frac{TN}{FP + TN} \quad (\text{II.58})$$

# Capítulo III

## Estado del arte

Hoy en día unos de los principales problemas que presenta la industria minera son la dureza y la sobre-estimación del tamaño de las rocas de donde se extrae el mineral, incorporando grandes gastos de energía para poder lograr de manera recomendable la extracción de éste mismo. Bajo este contexto y tomando énfasis en lo anteriormente mencionado se ha generado un estudio basado en la identificación de sobre-estimación de tamaño de mineral en procesos mineros, considerando que los sistemas informáticos han lanzado aplicaciones de reconocimiento de imágenes los cuales han permitido desarrollar algoritmos basados en el contenido de las imágenes, reconociendo formas y analizando su parecido.

En muchos países del mundo, el tamizado es el método más utilizado para la estimación del tamaño del mineral en los procesos mineros. Sin embargo, esta técnica no es útil cuando se requiere realizar una inspección del proceso en tiempo real. Es evidente que las técnicas mecánicas no permiten un mayor análisis de proceso y es por esta razón que las primeras investigaciones y aplicaciones apuntan a desarrollos de técnicas no mecánicas principalmente orientadas al uso de tecnologías basadas en el fotoanálisis de imágenes 2D [73]. Convirtiéndose el análisis de imágenes en un tema de investigación para la medición de partículas de rocas en diferentes aplicaciones tales como flujos gravitacionales, cintas transportadoras, pilas de rocas y en laboratorios [74].

Nuevas técnicas se han utilizado en los últimos treinta años debido al incremento computacional existentes, siendo la adquisición de imágenes y su procesamiento un nuevo potencial para medir el tamaño, y estimar la forma de las partículas. Los análisis en línea más tradicionales para estimar el tamaño de la roca están basados en la detección de bordes, segmentación, operadores morfológicos y uso de la transformada *watershed*. Alrousan y Masad [75, 76] evaluaron algunos de los métodos

de análisis de imágenes más utilizados para caracterizar la forma de las partículas, incluyendo el análisis de forma, angularidad y textura; en imágenes en escala de grises implementado el descriptor *wavelet*, segmentación como la erosión y dilatación.

Por lo tanto, se ha utilizado estas técnicas de procesamiento para estimar la distribución del tamaño del mineral mediante el análisis de imagen superficial de las pilas de partículas desde 1980. Donde King [77] propuso una medición simple para clasificar dos tipos de minerales, el de sobre-tamaño con análisis de imagen, por medio de la técnicas estereológicas y segmentación de partículas de interés, utilizando imágenes en escala de grises. Demostrando que el fotoanálisis es efectivo y preciso, lo cual fue útil para estudios posteriores como Petruk [78] el cual utilizó la segmentación de imágenes para automatizar la distribución de tamaños por medio de los niveles de gris que se presentaba en las imágenes de rayos X que presentaban las partículas de las rocas.

Sin embargo, existe un problema común para este tipo de algoritmos debido al proceso de segmentación de las imágenes [79]. Otra dificultad adicional, es la configuración de parámetros para este tipo de algoritmos que deben ser ajustados manualmente para cada tipo de situación donde se debe suponer que esas condiciones de borde no cambiarán en el tiempo. Algunos investigadores han intentado proponer nuevos procedimientos de segmentación como X. Wu y J.M. Kemeny [80] quienes desarrollaron un sistema automático de segmentación y detección de contornos de sombra para la separación de las partículas con la ayuda del cálculo de los gradientes que ellos generan, este método fue implementado para calcular la distribución del tamaño de los fragmentos de roca creados por voladura. Sin embargo este método presenta problemas por la superposición de las partículas y error de estimación de ellas para predecir un tamaño adecuado.

Es por ello que se generan nuevos estudios para dejar a un lado el error de superposición de las partículas y la implementación de nuevos métodos como Maerz [81] quien ocupa técnicas de fusión de zoom al igual minimizar los errores de perspectiva, visualizando imágenes casi en ángulo recto con la superficie de la roca que se va a medir. Mkwelo y Al-Thyabat [82, 83] utilizan para la detección de rocas con sobre tamaño la ayuda del cálculo del gradiente por medio de la transformada *Watershed*, en ella obtuvieron un *accuracy* de 89,9% con una base de datos de 20 imágenes, las cuales fueron trabajadas en escala de grises para detectar las regiones de interés. Al igual muchas empresas desarrollaron paquetes comerciales para su implementación industrial como [84, 85] donde utilizan algoritmos de segmentación y técnicas de procesamiento de imágenes para calcular en tiempo real las distribuciones del tamaño del mineral a partir de imágenes en escala de grises. Otros estudios trabajaron en errores de delineación de partículas con superficies 3D como [86, 87, 88, 89, 90] donde examinaron las segmentaciones tridimensionales de la superficies en pilas de rocas de laboratorio y determinaron qué regiones de la segmentación correspondían a

rocas completamente visibles, y cuales correspondían a rocas superpuestas o parcialmente visibles y otros como Thurley [91] ocuparon un perfil de Láser superficial 3D.

Nuevas ideas se han presentado como Liao [92] quien estableció un sistema de distribución de tamaños de partículas gruesas en línea implementando la separación del mineral con un motor de vibración electromagnético, con tolva y trayectoria de alimentación del material donde el procesamiento de la imagen fue implementado por la umbralización y dilatación de la imagen o también llamado matemática morfológica para contrarrestar los errores de superposición de las partículas. Zhang [93] propuso un nuevo algoritmo para segmentar las imágenes utilizando filtros lineales de múltiples escalas, construido con la matriz de *Hessian* y función gaussiana, donde este algoritmo obtiene una imagen con una mayor intensidad en los bordes para después ser segmentado por la transformada *Watershed* y Hamzeloo [94] integró para el cálculo de la distribución del tamaño de las partículas en concentrado de cobre análisis de componentes principales (PCA) y redes neuronales como clasificador en procesos industriales, donde fueron procesadas 21 muestras para la detección, el procesamiento fue con imágenes en escala de grises para realizar la separación entre las partículas.

La segmentación no deja de ser un tema de estudio para el cálculo de mineral con sobre-tamaño, donde las investigaciones trabajan para tener una mayor precisión con las rocas solapadas entre ellas. Zhang y Yang [95, 96] demuestran que sin importar el tiempo la segmentación es un tema investigativo, pero no solo la segmentación tomo fuerza para el problema planteado, debido que estos algoritmos realizan el análisis de búsquedas de límites que delimitan fragmentos de roca individuales, donde desafortunadamente los detectores de bordes simples tienen un desempeño pobre en estas tareas y son necesarios enfoques más sofisticados generando gastos computacionales y económicos.

Por lo tanto se han presentado metodologías de clasificación de imágenes para estimar el tamaño de las partículas de rocas que no dependa de la segmentación, como en los minerales también se pueden observar colores, formas y texturas, es allí donde el procesamiento de imagen basado en textura tomó importancia sin importar que se encuentre limitado en la industria principalmente a la dificultad de relacionar cálculos abstractos con características significativas y observables, pero las características de las texturas son los mejores parámetros para tener información valiosa de una imagen para distinguirla y clasificarla. Existen dos métodos principales para extraer estas características, ellos son el método de primer orden (Histogramas) y el método estadístico de segundo orden basado en la matriz de co-ocurrencia de nivel de gris que es el más ocupado en los procesos industriales [97]. Donde las características de la imagen a investigar se denominan textura, las cuales consisten conceptualmente en características tales como manchas, terminaciones

de línea y cruces de líneas [98]. Además, los métodos de extracción de características son diseñados para tolerar los problemas presentados en la segmentación como la calidad de las imágenes que a su vez, dependen de la iluminación y la configuración de la cámara.

Es por ello que el uso de estas características se ha ocupado en procesos de control inteligente basado en visión artificial en el procesamiento de minerales. Por ejemplo, Moolman y Kaartinen [99, 100] utilizan características de textura GLCM (Grey Level Co-occurrence Matrix) en los sistemas de control basados en visión artificial en celdas de flotación por espuma y estiman el tamaño promedio de las partículas distinguiendo los tipos de minerales en los sistemas de alimentación de mineral industrial [101, 102].

Para esta investigación se presentan aquellos estudios para el procesamiento de mineral y así estimar el tamaño de las partículas más relevantes, considerando a Jemwa [103] quien realizó un trabajo en la clasificación de carbón. El procesamiento de las imágenes fueron representadas como histogramas, para representar las características de texturas, el algoritmo implementado fue el de Varma y Zisserman al igual generaron un estudio de los métodos de filtros como uso de las texturas, y Bianconi [104] quien evaluó un amplio conjunto de descriptores de características de textura de color con el fin de clasificar granitos, comparando estrategias para integrar la textura y los datos de color, evidenciando descriptores que proporcionan alta precisión y bajo costo computacional.

Además de la matriz de co-ocurrencia, se pueden emplear otros descriptores de textura de escala de grises, como lo es el LBP (Local binary patterns) para la extracción de características de textura, demostrando que su eficiencia puede ser amplia en las aplicaciones industriales. Por otra parte, Kistner [15] realiza estudios para la clasificación de imágenes para la flotación de espumas de metales de platino industrial y partículas de carbón sobre una cinta transportadora para clasificar el mineral implementado como descriptores de características la matriz de co-ocurrencia y las representaciones complejas de wavelet multiescalar, la cual contaba con una base de total de 280 imágenes con un tamaño en píxeles de  $1136 \times 852$ . Mientras Amankwah [16] para la distribución de tamaño de partículas lo realiza mediante la unión de la segmentación (operaciones morfológicas) y la transformada *wavelet*. Con siete muestras diferentes de minerales con una base de datos de imágenes de 70 por cada una de las muestras. Donde cada una de ellas fue procesada bidimensionalmente en escala de grises.

La textura del mineral muestra ser un área de investigación, es por ello que Bull [105] propuso un método nuevo para delinear fragmentos de rocas por medio del descriptor *Haar*, extrayendo pequeños parches de la imagen para después implementar el clasificador SVM (Support Vector Machines), del mismo modo Perez [14] dividió cada imagen digital en subimágenes para extraer

información de la textura que estas brindaban con la ayuda del filtro de Gabor y la unión de la transformada *Watershed* con histogramas. Luego estas fueron clasificadas por una Máquina de Vectores de Soporte (SVM). Este proceso contó con tres bases de datos una de las cuales fueron subimágenes de  $64 \times 43$  píxeles de cinco tipos de mineral asignados a tres clases de molienda (blando, medio y duro). La segunda base subimágenes de  $60 \times 40$  píxeles y la tercera base de datos subimágenes de  $64 \times 48$ . En los experimentos se demostró que el clasificador SVM tiene mejores resultados que la red neuronal y que el análisis de textura de una imagen puede tener buenos resultados. En Zhang [106] no solo utilizó el clasificador SVM para su investigación, si no que hace una comparación de resultados buscando una mejor precisión, implementando redes neuronales y MLR (Regresión Lineal Múltiple) para predecir el contenido de cenizas en una correa transportadora utilizando como descriptor la utilización de características de histogramas.

Las investigaciones antes mencionadas hacen pie a la extracción de características de la textura del mineral, donde varía el tipo de descriptor a implementar en el proceso, ya sea transformando la intensidad de la imagen en cinco dominios de frecuencia como lo realiza Patel [11], quién utiliza la transformada discreta del coseno (DCT), transformada discreta de Fourier (DFT), la transformada discreta de Wavelet (DWT), la función de distribución acumulativa (CDF) y el filtro de Gabor o como Galdame [12] donde presenta un método basado en color y características en 3D para clasificar el mineral. El método ocupa imágenes de rango de color y láser, adquiridas en rocas en cintas transportadora, implementando los descriptores Gabor y LBP para después ser clasificados por el SVM.

Como se observó hay muchos métodos de aprendizaje automático de última generación los cuales utilizan descriptores de textura, para la clasificación de minerales. Es por ello que se busca un nuevo descriptor el cual sea robusto a cambios del sistema por la naturaleza del proceso minero, como la iluminación, rotación, entre otros.

Uno de ellos es el LBP propuesto por primera vez por Ojala [8], es uno de los descriptores más utilizados debido a su resistencia a los cambios de iluminación, al igual para calcular los cambios monótonos de la escala de grises de la imagen, la baja complejidad computacional y la capacidad de codificar detalles finos. LBP ha sido ampliamente estudiado en una gama de campos como análisis de caras, restos de fondo de vídeos, descripción de regiones de interés, clasificación de textura, etc. Un ejemplo de ello es el desarrollo de un sistema experto para la clasificación autónoma de baldosas de granito, donde se logró resultados de precisión de clasificación mayores al 94% con pocas características y un número limitado de muestras de entrenamiento, con la ayuda del clasificador SVM [104], al igual se ha demostrado un rendimiento superior en varios estudios comparativos.

Otro descriptor que tiene un alto rendimiento en textura es el HOG, el cual es utilizado en el procesamiento de imágenes con el fin de detectar objetos. Este método es invariante a las escalas de características y contextos de forma [107]. Este descriptor toma importancia en una conferencia en los años del 2005 sobre visión por computador y reconocimiento de patrones, por los investigadores del instituto Francés Navneet Dalal y Bill Triggs [38], en el cual se centraron en la detección de peatones en imágenes estadísticas, aunque desde entonces ampliaron sus pruebas para incluir la detección de humanos en vídeos, así a una variedad de animales comunes y vehículos en imágenes estáticas, detección e identificación de rostros. Mostrando ventajas claves sobre otros descriptores [108]. Estas ventajas son debido al modo de funcionamiento, el cual ocupa celdas locales, que le permiten ser invariantes para transformaciones geométricas y fotométricas, a excepción de la orientación a objetos.

Es por ello que este trabajo se propuso un método de textura basada en características de textura en minerales, comparando los descriptores resultantes de las características LBP y HOG, por tener un alto rendimiento al igual ser áreas nuevas en el procesamiento de imágenes como lo es el HOG. Buscando obtener resultados óptimos para el proceso establecido. Donde el rendimiento de clasificación fue con la ayuda del clasificador SVM.

# Capítulo IV

## Metodología

### 4.1. Metodología

La metodología utilizada corresponde a la descripción de los parámetros con los que se realizaron las pruebas en software para cada descriptor de características de mineral, indicando la base de datos implementada y la forma en que se realizó el entrenamiento de los clasificadores.

### 4.2. Introducción

En este capítulo, se describe la estrategia y metodología aplicada, para el desarrollo que permite la obtención de clasificación de los minerales con sobre-tamaño. Está compuesta por un conjunto de etapas que se muestra en el diagrama de bloques mostrados en la Figura 4.1, donde cada una de ellas es analizada en forma independiente, permitiendo cuantificar y analizar las ventajas y desventajas de su utilización.

Como se muestra en la Figura 4.1, la implementación de los experimentos consta de cinco pasos y el cual se inicia con la adquisición de las imágenes hasta la validación de los modelos propuesto por medio de un conjunto de métricas definidas en el marco teórico. En general, cada experimento considera una etapa de carga de las imágenes capturadas, para luego ser normalizadas en color y tamaño. Seguido a esto, cada imagen es convertida a un nuevo espacio de características utilizando los descriptores HOG y LBP. A partir de estos datos se construyen los nuevos modelos utilizando

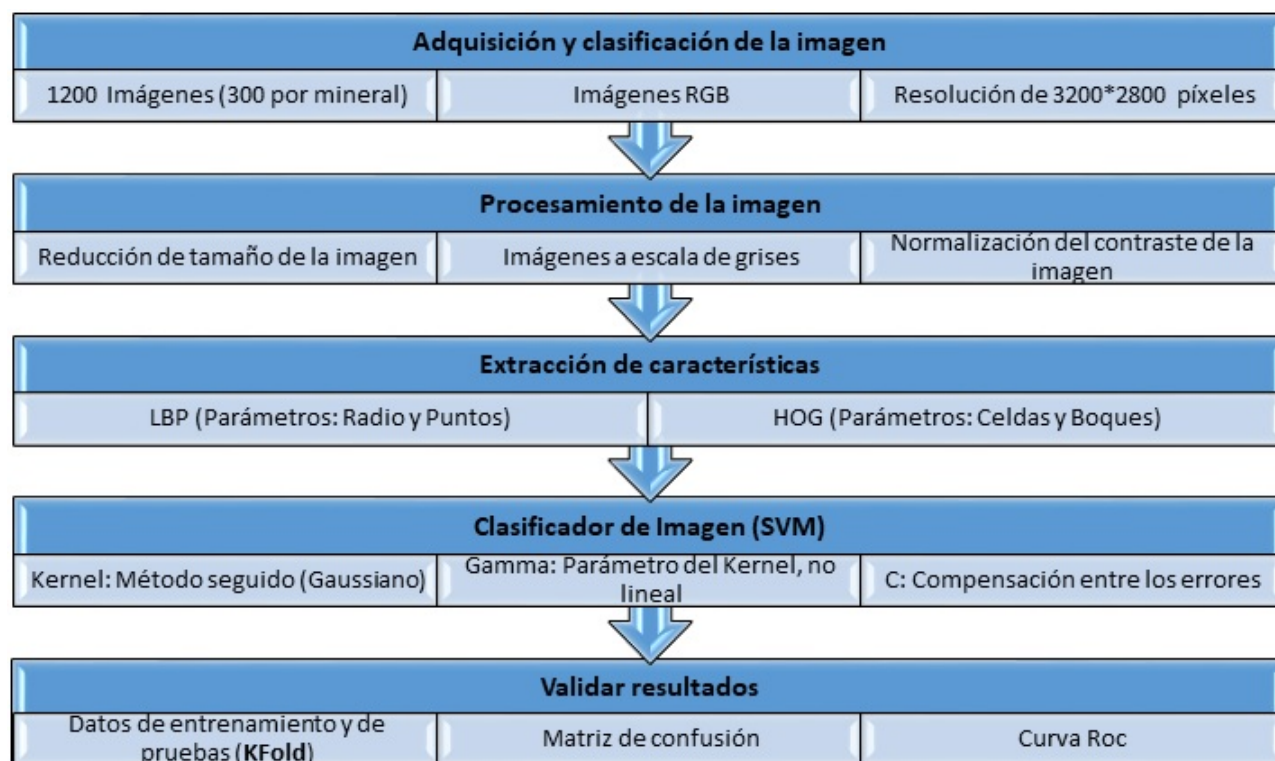


FIGURA 4.1: Pasos principales desde la adquisición de imágenes hasta la clasificación de rocas.

un clasificador SVM. Finalmente, con los resultados obtenidos, se hace un análisis detallado del rendimiento de cada una de las configuraciones definidas para cada uno de los experimentos.

Las diferentes configuraciones utilizadas servirán para poder comparar y validar los diferentes modelos propuestos en términos de eficiencia y que son detallados en las siguientes secciones.

### 4.3. Adquisición y clasificación de la Imagen

Para la evaluación de los algoritmos de detección de minerales con sobre-tamaño con imágenes, se utilizó una base de datos no publicada y nueva del Instituto Ceduc UCN (Universidad Católica del Norte) en Chile, esta base de datos consiste en imágenes RGB instantáneas (una por mineral) de cuatro tipos de minerales (Tamaño de mineral de cobre). Donde estas se han adquirido del departamento de Metalurgia de dicha institución las cuales fueron tomadas con una cámara digital (Sony Alpha 58) la cual tiene una resolución en torno a los 20 megapíxeles. Permitiendo tener imágenes de un máximo de  $5456 \times 3632$  píxeles. Las imágenes tomadas tienen una resolución fija cada una de  $3222 \times 2800$  píxeles y condiciones de iluminación variables. Hay 1,200 imágenes

con mineral, 300 Clase 1, 300 Clase 2, 300 Clase 3 y 300 Clase 4, mostradas en la Figura 4.2. En nuestro experimento, utilizamos las 1,200 imágenes, generando la base de datos con los cuatro tipos de mineral. Las cuales se puede acceder de manera gratuita, haciendo la solicitud al profesor Juan Bekios [109]. Sobre esta base de datos se realizaron pruebas para los descriptores de características LBP y HOG.

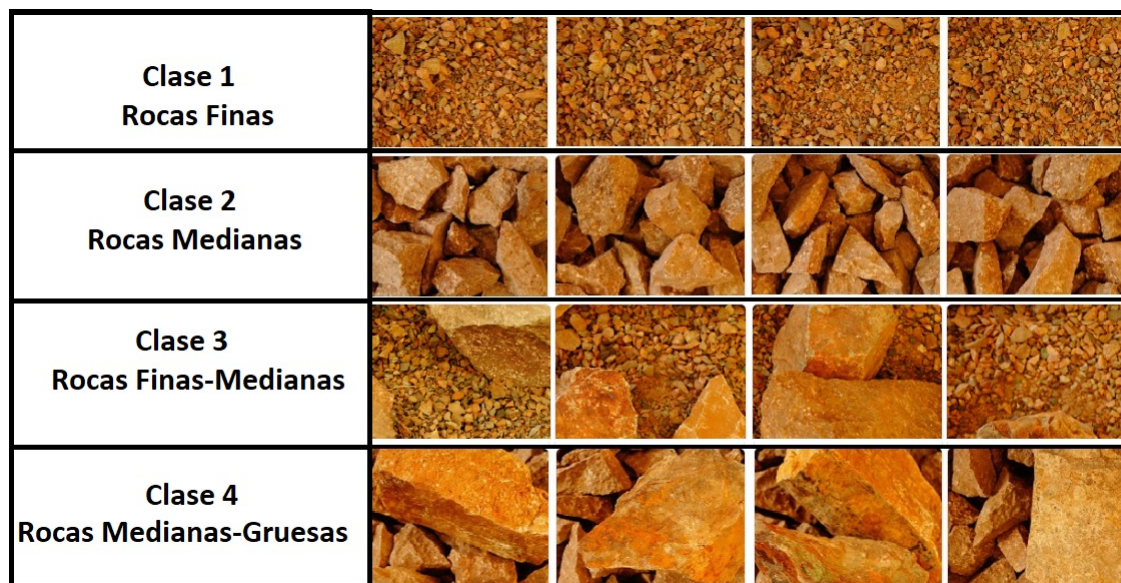


FIGURA 4.2: Clases de minerales ocupados en el proceso.

#### 4.4. Procesamiento de las Imágenes

Para la realización de las pruebas en los software, en primer lugar, se adaptaron las imágenes de la base de datos, reduciendo la dimensionalidad de la imagen, con el fin de mantener las propiedades de estas, y sus puntos característicos más definidos. Donde los valores implementados fueron imágenes cuadradas de ( $20 \times 20$ ,  $40 \times 40$ ,  $80 \times 80$ ,  $160 \times 160$ ,  $320 \times 320$ ,  $640 \times 640$ ,  $1280 \times 1280$  y  $2560 \times 2560$ ) píxeles. Posterior a esto, se reducen las señales de bajo nivel, ya que es considerado como ruido, para eliminar estas regiones de baja intensidad, se modificó a la escala de grises cada imagen. Por último para el procesamiento de la imagen se normalizó el contraste de cada una de las imágenes para dejarlas más homogéneas para el proceso. Este proceso es resumido en el Algoritmo 1.

## 4.5. Extracción de características de las Imágenes

Este proceso se dividió en dos etapas, debido que los descriptores contaban con parámetros diferentes. Los cuales se presentan a continuación.

**Cálculo del descriptor HOG:** El cálculo de histogramas de gradientes orientados (HOG) requiere generar imágenes con gradientes de forma horizontal y vertical, para obtener las matrices de magnitud y ángulo de cada píxel. Con ambas matrices se generan histogramas para distintas divisiones dentro de la imagen, las que posteriormente son concatenadas y normalizadas formando bloques, con los que finalmente se generan los descriptores de cada región. Este proceso se debe repetir para cada ventana análisis, como se explicó en el Capítulo 2.

El módulo de procesamiento de HOG requiere como entrada una imagen presente, la cual se encuentra en la base de datos obtenida. Ésta es accedida de forma secuencial generando gradientes para cada píxel de la imagen, excluyendo las filas y columnas de los extremos. Luego la imagen es dividida en celdas de ( $8 \times 8$ ,  $16 \times 16$ ,  $32 \times 32$ ,  $64 \times 64$ ,  $128 \times 128$ ) píxeles, generando un histograma de los gradientes correspondientes a cada una de ellas. Se tomó una variación de celdas con el fin de ver el comportamiento de cada uno de ellos buscando un mejor resultado en ellas.

Cada histograma generado posee 9 o 9 muestras posibles de ángulos en valor absoluto, esto quiere decir que cada muestra contiene un rango entre  $20^\circ$  y  $22,5^\circ$  cuya ocurrencia se genera según la magnitud del gradiente respectivo. El proceso de creación de la matriz de histogramas por celda se realizó sólo una vez por imagen.

Posteriormente, y para cada tamaño de análisis, se construyeron bloques de ( $1 \times 1$ ,  $2 \times 2$ ,  $3 \times 3$ ,  $64 \times 64$ ,  $128 \times 128$ ) celdas, esto corresponde a un problema de concatenación y normalización de histogramas, en el que se requiere crear una nueva matriz de bloques, cuyas dimensiones son el número de celdas menos uno. Para crear la matriz de bloques fue necesario indexar de forma correcta las celdas utilizadas en cada bloque, ya que, en tamaños distintos al original, las celdas contiguas no son parte del mismo bloque. Por esto, para construir los bloques se necesitó conocer el factor de escala, respecto al tamaño inicial, en que se realiza el análisis, ya que en esa cantidad de celdas está separada la siguiente celda perteneciente al bloque. Una vez realizada la copia de datos desde la matriz de celdas a la de bloques, cada bloque fue normalizado, expresando el resultado. Con esto se generaron todos los bloques posibles para un tamaño de ventana determinado en la imagen, posteriormente

estos bloques fueron concatenados para poder clasificarlos.

Todo el proceso anterior se muestra en la Figura 2.17, donde se indica la entrada de datos, el cálculo de los gradientes de cada píxel y el proceso de creación de bloques, que corresponde a la salida de este módulo de procesamiento.

El descriptor HOG implementado en el trabajo se puede observar en el Algoritmo 1, el cual es una función de *Scikit-image* (Image processing in python) mostrada en el Capítulo 2.

---

**Algoritmo 1** Función para el cálculo del descriptor *HOG* en una imagen utilizando la librería *Scikit-image* escrita en Python.

---

**Entrada:**

*image*. Imagen del mineral en escala de grises.

*orientations*. Orientaciones que se trabaran donde puede ser 8 o 9.

*pixelscell*. Tamaño de la celda compuesta por el número de píxeles

*cellblock*. Cantidad de celdas que conformaran el bloque.

**Salida:**

*HOG*. Vector resultante que contiene el descriptor *HOG* de una imagen.

1:  $HOG = \text{hog}(\text{imagen}, \text{orientations}=\text{N}, \text{pixelscell}=(\text{N}, \text{N}), \text{cellsblock} = (\text{N}, \text{N}))$

---

**Cálculo del descriptor LBP:** Como se explicó en el Capítulo 2, el descriptor LBP es un filtro aplicado a cada píxel de la imagen a procesar, que depende de un radio y un número de puntos fijos. Así, para este método de extracción de característica se tienen estos dos parámetros que se deben ajustar. Es por ello que la metodología ocupada fue la variación de estos dos parámetros con el fin de tener una mejor posición en el proceso. Los radios trabajados fueron (2, 3, 4, 5, 6, 7, 8, 9) y número de puntos (2, 4, 8, 16, 32, 64, 128, 256). Los cuales permitieron la obtención de una etiqueta que representa una micro textura asociada a cada vecindad del píxel.

En el enfoque propuesto en este trabajo, el procesamiento se realiza en cada región por separado, esto es transparente para el cálculo del operador LBP, ya que sólo se debe indicar el radio con el incremento de los posibles puntos fijos que se pueden presentar en la descripción. Donde para el uso de un radio de dos se ocupó una cantidad de puntos homogéneos a ellos los cuales fueron dos y cuatro que componían el área de trabajo de la circunferencia, así se realizó para los distintos tipos de radios y su incremento en puntos.

El descriptor *LBP* se puede observar en el Algoritmo 2, el cual también es una función de *Scikit-image*.

---

**Algoritmo 2** Función para el cálculo del descriptor *LBP* en una imagen utilizando la librería *Scikit-image* escrita en Python.

---

**Entrada:**

*image*. Imagen del mineral en escala de grises.

*npoints*. Número de puntos de referencia que se utilizaron en la circunferencia.

*radius*. Valor del radio que ocupará la circunferencia.

**Salida:**

*LBP*. Vector resultante que contiene el descriptor *LBP* de una imagen.

1:  $LBP = \text{lbp}(\text{image}, \text{npoints}, \text{radius})$

---

## 4.6. Clasificación de imágenes

El clasificador SVM fue el encargado de identificar si un vector de características, generado por los descriptores LBP o HOG para los cuatro tipos de minerales, presenta o no condiciones de sobretamaño de mineral. Para esto, el clasificador combina el vector del descriptor generado con una etiqueta que pertenece cada clase de mineral, como ejemplo si el descriptor pertenece al primer tipo de mineral este tomara una etiqueta o valor de uno como se mostró en la Figura 4.2.

Como se mencionó en la sección anterior, el cálculo del LBP y HOG generan descriptores para todas las posibles regiones según las condiciones que este requiera, presentes en la imagen. Enviando cada una de ellas al clasificador. Por esto el clasificador recibe como entrada un vector de la imagen analizada más la clase de éste, en ese caso particular este vector corresponde a un tipo de mineral independientemente del descriptor que se haya implementado.

Generando los valores de clasificación que debe ser comparado con los umbrales de flujo de implementación del clasificador SVM mostrados en el Algoritmo 3.

En el cual hay parámetros que se debieron identificar y clasificar para obtener un óptimo resultado, siendo estos el *Kernel*, *C* y *Gamma*.

**Kernel:** El cuál es la función matemática utilizada para la transformación de la clasificación para sistemas no lineales como lo es el trabajo realizado. Estas funciones resuelven el problema de clasificación trasladando los datos a un espacio donde el hiperplano solución es lineal y, por tanto, más sencillo de obtener. Una vez conseguido, la solución se transforma, de nuevo, al espacio original mostrado en el capítulo 2. Es por ello que se ocupó el *Kernel* que tiene por defecto el sistema, el cual es *rbf* siendo este una función de base radial.

---

**Algoritmo 3** Cálculo de parámetros  $\gamma$  y  $C$  para el clasificador *SVM* con diferentes configuraciones de descriptores *LBP* y *HOG* aplicadas a la base de datos de imágenes de minerales.

---

**Entrada:**

$f = \{data_i, label_i\}_{i=1}^N$ . Donde  $N$  es el número total de imágenes de la base de datos,  $data_i$  representa el descriptor *HOG* ó *LBP* aplicado a cada imagen y  $label_i$  valor de clase que pertenece a cada imagen.

**Salida:**

$clf_{mejor}$ . Mejor clasificador *SVM* para el descriptor *HOG* ó *LBP*.

```

1:  $dic \leftarrow \mathbf{Leer}(f)$ 
2:  $X \leftarrow dic['data']$     ▷ Matriz de imágenes. Cada fila contiene los descriptores LBP o HOG de una imagen.
3:  $y \leftarrow dic['labels']$     ▷ Vector de valores clase. Valor clase que corresponde a cada fila de la matriz  $X$ 
4:  $m_{mejor} \leftarrow 0$ 
5: para  $\gamma \in \{0,1;0,001;0,00001;0,000001\}$  hacer
6:   para  $C \in \{1;10;100;1000\}$  hacer
7:      $clf \leftarrow SVM(\gamma,C)$ 
8:      $clf.entrenar(X,y)$ 
9:      $m \leftarrow calcularMetrica(clf)$ 
10:    si  $m > m_{mejor}$  entonces
11:       $m_{mejor} \leftarrow m$ 
12:       $clf_{mejor} \leftarrow clf$ 

```

---

**Gamma ( $\gamma$ ):** Es una función kernel, lo que nos facilita encontrar los subespacios que puedan diferenciar los puntos en el espacio y también nos permite añadir mayor complejidad a la hora de separar observaciones. Para el entrenamiento se utilizaron los valores  $1e - 1$ ,  $1e - 3$ ,  $1e - 5$  y  $1e - 6$  para buscar el mejor valor para este parámetro.

**C:** Ayuda controlar la compensación entre los errores de entrenamiento. El cual permite que el modelo de clasificación tenga unos márgenes blandos, con el fin de ser flexibles en ciertas muestras. Siendo  $C$  el peso que le da a cada una de las observaciones, a la hora de clasificar un mayor coste implicaría un mayor peso de una observación y el *SVM* será más estricto. Es por ello que se utilizaron para el entrenamiento los valores 1, 10, 100 y 1000 con el fin de obtener el mejor valor posible en los resultados.

Se aplicó el algoritmo de máquina de vector de soporte (*SVM*) de las cuatro clases de minerales propuestos por la base de datos.

Este clasificador se mostrará en el Algoritmo 3, el cual es una función de *scikit-learn*, donde  $\gamma$  es el primer parámetro a trabajar después de tener tanto el descriptor como la clase de éste a clasificar,

seguido de cada uno de los valores propuestos por  $C$ . Nos permite identificar el mejor clasificador por medio de una matriz la cual tiene el valor estimado como mejor precisión por cada uno de los procesos.

## 4.7. Validación de los resultados

Para ver los resultados de los experimentos se utilizó la validación cruzada o *cross-validation* por la naturaleza del trabajo, al igual por su mayor objetividad en las pruebas realizadas.

En la cual se ocuparon cinco subconjuntos  $K = 5$ , donde el proceso para la obtención de los resultados tiene como etapa inicial el descriptor calculado con su respectiva etiqueta o clase separada por los subconjuntos estimados. Seguido de la clasificación con los parámetros ideales del SVM, el cual permiten una mejor eficiencia en la respuesta del sistema, esta representación se mostrará en la Figura 4.3.

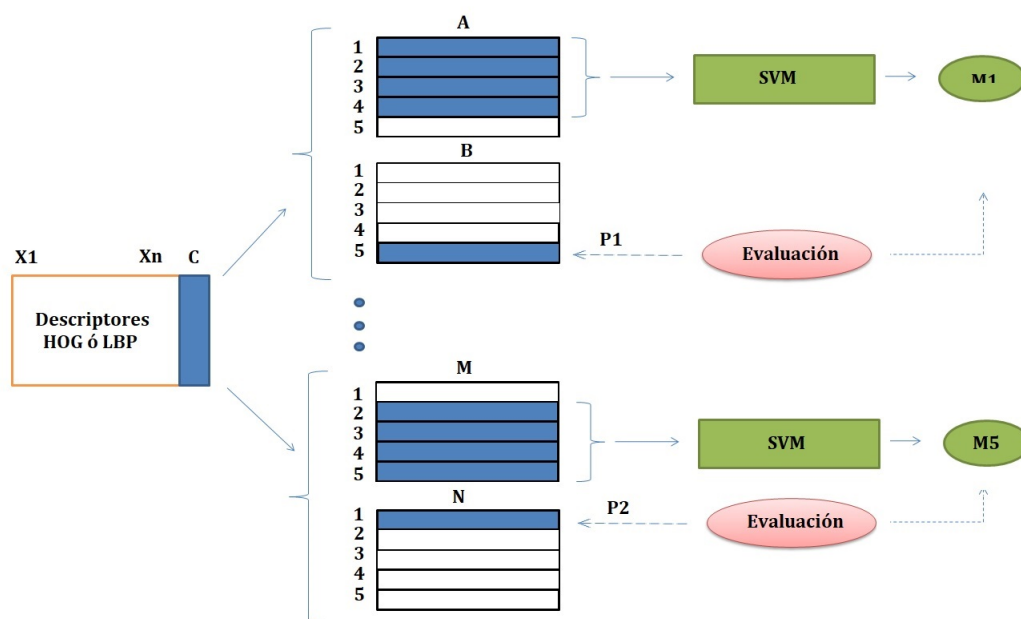


FIGURA 4.3: Método de estimación basado en  $k$  rodajas.

El método de la validación cruzada se puede observar en el Algoritmo 4, el cual también es una función de *Scikit-image*. La cual tiene la misma característica que el algoritmo de clasificación, la diferencia es la división de los subconjuntos para entrenar.

---

**Algoritmo 4** Cálculo de la validación *k-fold cross validation* con las condiciones ideales de los parámetros del clasificador y del descriptores *LBP* y *HOG* aplicadas a la base de datos de imágenes de minerales.

---

**Entrada:**

$f = \{data_i, label_i\}_{i=1}^N$ . Donde  $N$  es el número total de imágenes de la base de datos,  $data_i$  representa el descriptor *HOG* ó *LBP* aplicado a cada imagen y  $label_i$  valor de clase que pertenece a cada imagen.

$\gamma$  ( $\gamma$ ). Es un valor utilizado por el *kernel* RBF, calculado empíricamente para cada descriptor.

$C$ . Valor de coste de las observaciones, calculado empíricamente para cada descriptor.

**Salida:** *reales*, *predichas*. Resultados de clasificación calculada por el clasificador *SVM* para el descriptor *HOG* ó *LBP*.

- 1:  $dic \leftarrow \text{Leer}(f)$
  - 2:  $X \leftarrow dic['data']$      $\triangleright$  **Matriz de imágenes.** Cada fila contiene los descriptores *LBP* o *HOG* de una imagen.
  - 3:  $y \leftarrow dic['labels']$      $\triangleright$  **Vector de valores clase.** Valor clase que corresponde a cada fila de la matriz  $X$ .
  - 4:  $skf \leftarrow \text{StratifiedKFold}(y, K)$      $\triangleright$  **División de las muestras.** Se crean  $K$  conjuntos de entrenamiento y prueba,  $(\{train_i, test_i\}_{i=1}^k)$ , para cada uno de los experimentos.
  - 5: **para**  $\{train_i, test_i\} \in skf$  **hacer**
  - 6:     $clf \leftarrow \text{SVM}(\gamma, C)$      $\triangleright$  **Configuración clasificador SVM.**
  - 7:     $clf.entrenar(X[train_i, :], y[train_i])$      $\triangleright$  **Entrenamiento.**
  - 8:     $predichas \leftarrow clf.prededir(X[test_i, :])$      $\triangleright$  **Predicción.**
  - 9:     $reales \leftarrow y[test_i]$
  - 10:  $mc \leftarrow clf.obtenerMatrizDeConfusion(reales, predichas)$      $\triangleright$  **Matriz de confusión.**
  - 11:  $metricas \leftarrow calcularMetricas(mc)$      $\triangleright$  **Calcular tasa de acierto, precisión, entre otras.**
-

# Capítulo V

## Resultados Experimentales

En este capítulo se muestra una comparación de los resultados en términos de la eficiencia de clasificación medida a través de un conjunto de métricas donde destaca la tasa de acierto, precisión y *recall* (sensibilidad) entre otros. También se hace un breve análisis de rendimiento potencial para las etapas de entrenamiento y pruebas para cada modelo propuesto.

Los resultados obtenidos utilizando los dos descriptores LBP y HOG fueron realizados bajo las mismas condiciones, la cual fue un conjunto de datos de 4 categorías que difieren en la proporción de las muestras del mineral. Cada categoría contiene 300 muestras, las cuales se dividieron en conjuntos equilibrados de entrenamiento y prueba.

### 5.1. Descriptor LBP

Para los experimentos desarrollados con el descriptor LBP, se utilizó la metodología propuesta en el Capítulo IV. Para cada uno de los experimentos se utilizó la base de datos del *Instituto CEDUC UCN*. Se configuraron los parámetros de **radio** y **número de puntos** del descriptor LBP a través de validación cruzada. Finalmente, y una vez obtenidos los mejores parámetros para el descriptor LBP se realiza un conjunto de experimentos para obtener los mejores parámetros para el clasificador SVM.

### 5.1.1. Configuración del experimento

La configuración de parámetros para cada una de las etapas de los experimentos, ver Figura 4.1, fue la siguiente:

1. **Adquisición y clasificación de la imagen:** Base de datos Instituto CEDUC UCN. Cuatro tipos de clases: Rocas Finas, Rocas Medianas, Rocas Finas-Medianas y Rocas Medianas-Gruesas, con un total de 1200 imágenes en total distribuidas homogéneamente en las clases.
2. **Procesamiento de la imagen:** Las imágenes fueron normalizadas en intensidad. También, se re-escalaron a diferentes tamaños en *pixels* como  $(20 \times 20, 40 \times 40, 80 \times 80, 160 \times 160, 320 \times 320, 640 \times 640, 1280 \times 1280$  y  $2560 \times 2560)$  para realizar diferentes pruebas.
3. **Extracción de características:** Para la extracción de características utilizando el descriptor LBP fue necesario ajustar dos parámetros. El **radio**, donde se probaron valores entre 2 y 9. Para el **número de puntos** se probaron valores entre 2 y 256.
4. **Clasificación de la imagen:** Para el clasificador se ajustaron los valores de  $\gamma$  y  $C$ . Los valores de  $\gamma$  fueron para el procesp fueron 0, 1, 0, 001, 0, 00001, 0, 000001 y los valores de  $C$  fueron 1, 10, 100 1000.

TABLA 5.1: Precisiones altas por cada uno de los radios entrenados por el descriptor LBP.

Radios	2	3	4	5	6	7	8	9
Tasa de Aciertos %	89.2 (+/- 4.2)	89.9 (+/- 4.0)	<b>90.8 (+/- 5.8)</b>	90.4 (+/- 5.6)	90.7 (+/- 3.8)	90.6 (+/- 2.9)	88.6 (+/- 1.4)	90.6 (+/- 4.3)
Error %	10.8	10.1	<b>9.2</b>	9.6	9.3	9.4	11.4	9.4
Tamaño de la Imagen	80	80	<b>160</b>	160	160	320	160	320
Numero de Puntos	16	16	<b>16</b>	16	16	8	16	8
Clasificador SVM								
$\gamma$	1e-05	1e-05	<b>1e-06</b>	1e-06	1e-06	1e-06	1e-06	1e-06
$C$	10	10	<b>10</b>	10	10	10	10	10

### 5.1.2. Validación de los Resultados

La Tabla 5.1 muestra la precisión de clasificación de los mejores resultados por cada uno de los radios implementados en el entrenamiento del descriptor LBP para los distintos tipos de mineral,

estas van acompañadas de la tasa de aciertos, el error para una clasificación ideal, el tamaño que se ocupó en la imagen, el número de puntos a utilizar para ello y los parámetros de entrenamiento del clasificador SVM. En donde la precisión promedio para los ocho tipos de radios utilizados en la base de datos de prueba alcanzó el 90,1 %. Al mismo tiempo la mayor precisión utilizando este descriptor fue 90,8 % con una varianza  $+/- 5,8$  y un radio de 4. Lo cual no se encuentra distante al promedio dado.

Por otro lado, para el tamaño de la imagen se obtuvo un valor de 160 píxeles representando el 50 % de los tamaños evaluados y para el número de puntos utilizados, igual a 16, se obtiene el 80 % de coincidencia para los distintos radios.

La Tabla 5.2 muestra el comportamiento del descriptor con mayor precisión de acuerdo a la cantidad de puntos ocupado en el entrenamiento del vector a implementar versus el tamaño de la imagen del mineral. Observando que a mayor tamaño de la imagen y mayor número de puntos, la precisión disminuye. A menor tamaño de imagen y mayor número de puntos la precisión aumenta; sin embargo para un tamaño de imagen igual a 160 la precisión es mayor con el número de puntos promedio.

TABLA 5.2: Resultados obtenidos para un radio definido en 4, un aumento del tamaño de la imagen y número de puntos.

Radio 4	Tamaño de las Imágenes									
	40		80		160		320		640	
4	74.1(+/-3.9)		62.2(+/-1.8)		<b>83.8(+/-4.7)</b>		80.6(+/-2.8)		57.8(+/-5.9)	
	C=10	$\gamma = 0,001$	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 06$
8	65.0(+/-5.8)		81.0(+/-2.5)		<b>89.3(+/-4.5)</b>		82.6(+/-7.8)			
	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 05$	C=1	$\gamma = 1e - 06$		
16	75.8(+/-2.6)		88.1(+/-3.2)		<b>90.8(+/-5.8)</b>		66.3(+/-3.1)		41.6(+/-1.6)	
	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$
32	85,2(+/-1,5)		89,2(+/-4,6)		<b>81,2(+/-6,0)</b>		54,3(+/-5,6)		26,6(+/-2,5)	
	C= 10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$
64	88,3(+/-5,5)		46,8(+/-6,6)		<b>26,7(+/-1,8)</b>		31,4(+/-3,1)			
	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$		
128	85,3(+/-1,2)		50,8(+/-3,5)		<b>26,6(+/-1,2)</b>		39,0(+/-1,6)		25,1(+/-1,0)	
	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 05$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 1e - 06$	C=10	$\gamma = 0,1$

Para tener una interpretación ideal del sistema, se realizó una matriz de confusión por cada uno de las muestras, con el fin de observar otras variables distintas a la precisión como la sensibilidad, especificidad, y *accuracy*.

Como se puede apreciar en la Tabla 5.3 el descriptor LBP tiene en promedio un *accuracy* de 95,63 %, siendo este alto al momento de clasificar los distintos tipos de minerales. Al igual puede discriminar los verdaderos negativos en óptimas condiciones con un porcentaje promedio de 97,1 % (Especificidad). Por otro lado se tiene un promedio bajo en precisión y sensibilidad

TABLA 5.3: Análisis de desempeño de clasificación con el descriptor LBP por cada uno de los 5 Fold.

	Sensibilidad (%)	Especificidad (%)	Precisión (%)	Accuracy (%)
<i>Fold 1</i>	92,70	96,72	89,58	95,41
<i>Fold 2</i>	95,86	99,02	97,06	98,22
<i>Fold 3</i>	88,29	96,14	88,28	94,19
<i>Fold 4</i>	92,20	97,29	91,66	95,93
<i>Fold 5</i>	88,64	96,33	88,75	94,43
<b>Promedio</b>	<b>91,53</b>	<b>97,1</b>	<b>91,06</b>	<b>95,63</b>

indicando que el sistema en la detección de los verdaderos positivos no es óptimo y presenta errores los cuales no son viables en la detección ideal del proceso.

Estos resultados son favorables en la clasificación de los distintos tipos de minerales, para el descriptor, cabe mencionar el Fold 2, donde se aprecian los mejores resultados los cuales toma una sensibilidad del 95,86%, especificidad del 99,02%, precisión del 97,06% y *accuracy* del 98,22%. Con estos datos se puede decir que el sistema es capaz de clasificar los tipos de minerales propuesto en la investigación con altos resultados.

## 5.2. Descriptor HOG

Los experimentos para el descriptor HOG, se realizaron con la misma metodología ocupada por el descriptor LBP. Donde se diferencia del anterior en la configuración de los parámetros del descriptor, como lo es el tamaño de la **celda** en píxeles y el tamaño del **bloque** configurados a través de la validación cruzada. Después de obtener los mejores parámetros del descriptor HOG se realizan los experimentos con el fin de identificar los parámetros ideales para el clasificados SVM.

### 5.2.1. Configuración del experimento

La configuración de los parámetros tiene las mismas características del descriptor LBP, las cuales se pueden ver en la ver Figura 4.1, el cual es el siguiente:

1. **Adquisición y clasificación de la imagen:** Base de datos Instituto CEDUC UCN. con los cuatro tipos de clases.

2. **Procesamiento de la imagen:** La normalización de las imágenes en intensidad y la reducción de tamaño a diferentes dimensiones de *pixels* como ( $20 \times 20$ ,  $40 \times 40$ ,  $80 \times 80$ ,  $160 \times 160$ ,  $320 \times 320$ ,  $640 \times 640$ ,  $1280 \times 1280$  y  $2560 \times 2560$ ) para realizar diferentes pruebas.
3. **Extracción de características:** Para la extracción de características utilizando el descriptor HOG fue necesario ajustar dos parámetros. La **celda**, donde se probaron valores en entre  $8 \times 8$  hasta  $128 \times 128$ . Para el tamaño de **bloques** se probaron valores entre  $1 \times 1$  hasta  $128 \times 128$  con incrementos exponenciales al valor inicial.
4. **Clasificación de la imagen:** Para el clasificador se ajustaron los valores de  $\gamma$  y  $C$ . Los valores de  $\gamma$  para el entrenamiento fueron 0, 1, 0,001, 0,00001, 0,000001 y los valores de  $C$  fueron 1, 10, 100 1000.

## 5.2.2. Validación de los Resultados

La Tabla 5.4 muestra la precisión de clasificación de los mejores resultados por cada una de las celdas implementadas en el descriptor HOG, acompañadas de los parámetros mencionados anteriormente en el descriptor LBP, más el tamaño del bloque; en donde para un tamaño de celda igual a  $32 \times 32$  píxeles, se obtiene la mejor precisión del clasificador de 96,1 % con un 3,9 margen de error a lo ideal y un tamaño de la imagen igual a  $1280 \times 1280$  píxeles.

Por otro lado el tamaño del bloque predominante es 1 el cual corresponde al tamaño de la celda.

TABLA 5.4: Resultados para el tamaño de las celdas y bloques utilizados en la clasificación del descriptor HOG

Tamaño de la Celda	8	16	32	64	128
Tasa de Aciertos %	94.3(+/-4.3)	95.3(+/-2.5)	<b>96.1(+/-1.7)</b>	91,9(+/-4,2)	91,7 ( +/- 7.2)
Error %	5.7	4.7	<b>3,9</b>	8.1	8.3
Tamaño de la Imagen	320	1280	<b>1280</b>	2560	2560
Tamaño del Bloque	1	1	<b>1</b>	2	1

Clasificador SVM					
$\gamma$	0.01	0.001	<b>0.001</b>	0.01	0.001
c	10	100	<b>100</b>	10	100

La Tabla 5.5 muestra las precisiones evaluadas para el mejor tamaño de celda obtenido en el momento de clasificar los distintos tipos de mineral; visualizando las principales variables que afectan el proceso siendo estas el aumento de tamaño de imagen v/s el número de bloques a utilizar. Observando que el incremento de la imagen aumenta la precisión del clasificador hasta llegar al tamaño máximo de entrenamiento de 2560 píxeles, no así al incrementar el tamaño del bloque ya que la predicción va disminuyendo.

TABLA 5.5: Resultados globales con una tamaño de celda de 32 píxeles, con un incremento de tamaño y bloques.

Tamaño de la Celda: 32	Tamaño de la Imagen													
Tamaño del Bloque	40		80		160		320		640		1280		2560	
1*1	52.4(+/-5.2)		63.1(+/-5.1)		75.0(+/-4.7)		88.2(+/-3.9)		93.6(+/-3.1)		<b>96.1(+/-1.7)</b>		95.7(+/-1.1)	
	C= 100	$\gamma= 0.1$	C= 100	$\gamma= 0.1$	C= 100	$\gamma= 0.1$	C= 10	$\gamma= 0.1$	C= 10	$\gamma= 0.1$	C= 100	$\gamma=0.001$	C= 100	$\gamma=0.001$
2*2			61.6(+/-3.7)		80.0(+/-6.4)		87.3(+/-5.7)		93.0(+/-1.6)		<b>93.7(+/-2.5)</b>		94.9(+/-1.4)	
			C= 100	$\gamma= 0.1$	C= 100	$\gamma= 0.1$	C= 10	$\gamma= 0.1$	C= 10	$\gamma= 0.1$	C= 100	$\gamma= 0.1$	C= 100	$\gamma= 0.1$
3*3					78.5(+/-7.9)		84.7(+/-4.0)		91.6(+/-2.1)		<b>93.1(+/-1.6)</b>		93.9(+/-0.7)	
					C= 100	$\gamma= 0.1$	C= 10	$\gamma= 0.1$	C= 100	$\gamma= 0.1$	C= 100	$\gamma=0.001$	C= 100	$\gamma=0.001$
4*4					73.5(+/-5.0)		83.5(+/-6.8)		89.9(+/-2.1)		<b>93.1(+/-1.7)</b>			
					C= 100	$\gamma= 0.1$	C= 100	$\gamma= 0.1$	C= 100	$\gamma= 0.1$	C= 10	$\gamma= 0.1$		

Al igual que el LBP se realizaron las matrices de confusión para el descriptor HOG para cada uno de los 5 *Fold*, con el fin de observar el comportamiento de la clasificación de las distintas clases de mineral.

La Tabla5.6 muestra el comportamiento de las tasas de aciertos que se lograron al momento de clasificar los tipos de minerales, los cuales se tiene un mejor resultado que el descriptor anterior (LBP). Obteniendo una *accuracy* de 98,17% y especificidad de 98,71% al momento de clasificar las clases con verdaderos negativos. Con una sensibilidades de 96,62% y precisión de 96,15%. Siendo estos resultados confiables al momento de identificar los verdaderos positivos en la clasificar los distintos tipos de minerales.

TABLA 5.6: Análisis de desempeño de clasificación utilizando el descriptor HOG.

	Sensibilidad (%)	Especificidad (%)	Precisión (%)	Accuracy (%)
<i>Fold 1</i>	94,61	98,19	94,58	97,29
<i>Fold 2</i>	96,32	98,76	96,25	98,12
<i>Fold 3</i>	95,93	98,61	95,83	97,91
<i>Fold 4</i>	97,12	98,61	95,89	98,22
<i>Fold 5</i>	97,11	98,89	96,66	98,43
<b>Promedio</b>	<b>96,62</b>	<b>98,71</b>	<b>96,15</b>	<b>98,17</b>

### 5.3. Evaluación del rendimiento de los modelos LBP y HOG

Con el propósito de analizar el rendimiento de nuestro método, se calculan las matrices de confusión para realizar en detalle el error de cada una de las clases (etiquetas) de mineral, debido que la precisión no es una métrica confiable por ser un valor global de los resultados y todo clasificador está sujeto a la posibilidad de cometer errores.

Es por ello que se realizó la implementación de las curvas ROC y AOC por la capacidad de darnos el diagnóstico detallado de discriminar los estados de las clases, mostrando los umbrales de decisión de ellos.

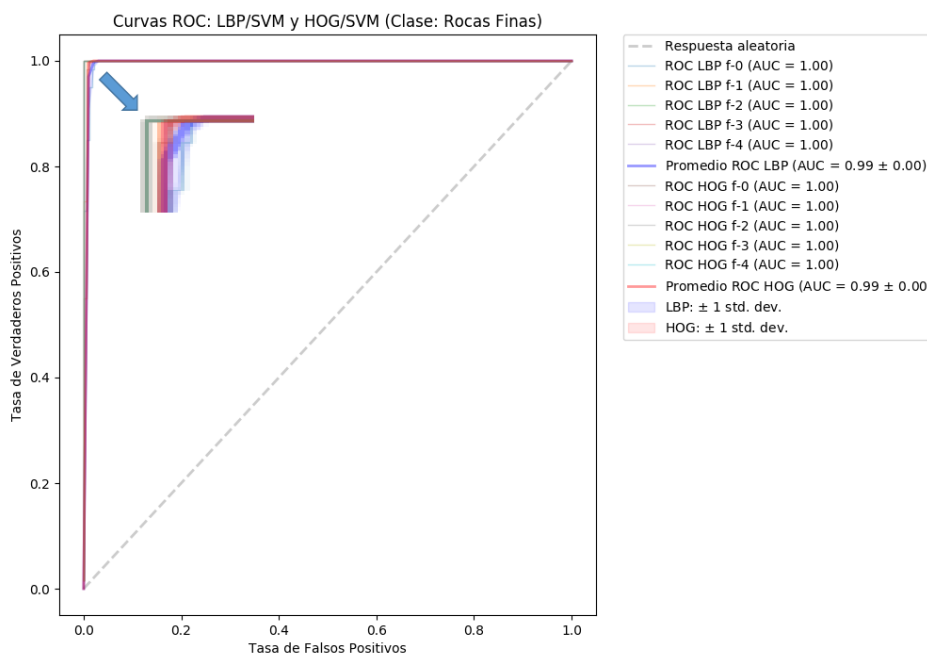


FIGURA 5.1: Curva ROC para la clase "Rocas Finas"

### 5.3.1. Eficiencia de los clasificadores

Para medir el rendimiento de cada una de las etiquetas se realizaron dos experimentos. El primero, que utiliza los mejores los parámetros de configuración para el descriptor LBP (parámetros) y SVM ( $\gamma=0.00001$ ,  $C=10$ ) y se realiza un entrenamiento en base a una validación *5 Fold*. A partir de cada *fold*, se calcula la probabilidad de cada valor clase y se construye una curva ROC para cada clase para poder comparar los resultados a nivel local, por cada *fold*, y a nivel global, y que es calculado a través del promedio de todos los *fold*. La métrica utilizada para la evaluación de los resultados es la tasa de verdaderos positivos contra la tasa de falsos positivos. También, se calcula la métrica AUC que es el área bajo la curva ROC (donde un mayor AUC indica un mejor desempeño del clasificador). El segundo experimento, se realiza bajo las mismas condiciones del primero pero se utilizan los mejores parámetros de configuración para el descriptor HOG (parámetros) y SVM ( $\gamma=0.00001$ ,  $C=10$ ).

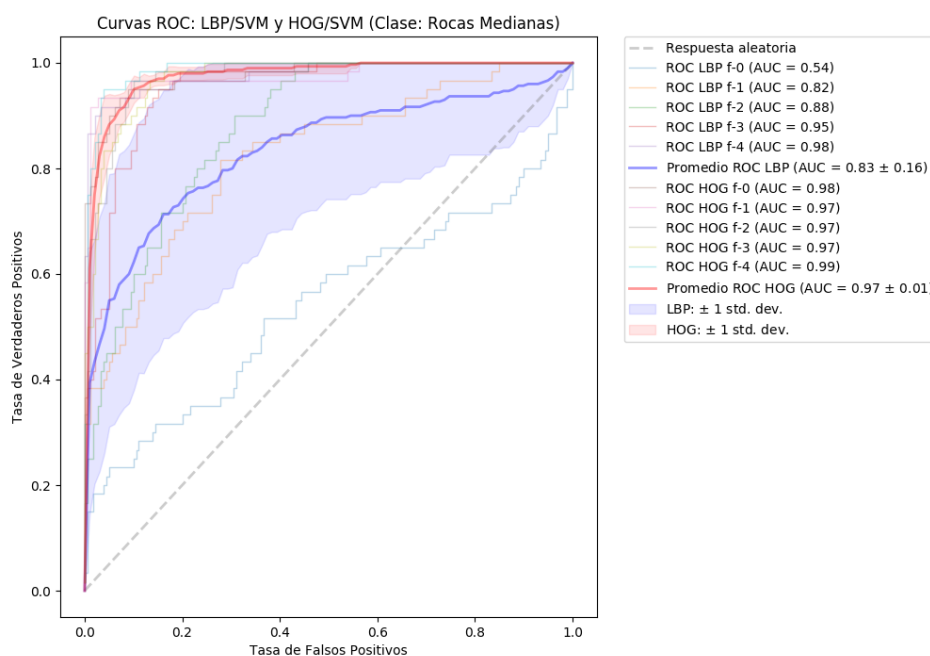


FIGURA 5.2: Curva ROC para la clase "Rocas Medianas"

Los resultados de cada clase y valores obtenidos en las métricas se discuten a continuación:

**Clase "Rocas Finas"** La clase presenta uno de los mejores rendimientos obtenidos para ambos experimentos con valores de AUC (Área bajo la curva) de 99% de área y un 1% de varianza. Esto indica que ambos descriptores tienen un desempeño similar y alto poder discriminante que poseen ambos descriptores, ver Figura 5.1.

**Clase "Rocas-Medianas"** El descriptor que presenta un alto rendimiento obtenido para la clasificación de este tipo de mineral fue el HOG, con un valor de AUC de 97% de área y un 1% de varianza. Superando al LBP, debido que este alcanzó un 83% y una alta varianza del 16%. Esto nos muestra que el descriptor HOG tiene un desempeño óptimo y alto poder discriminante al momento de clasificarlo, ver Figura 5.2.

La Figura 5.2 nos permite observar la dispersión de precisión obtenidos mediante los *5-fold* para los dos descriptores para la clasificación de este tipo de mineral, la precisión del LBP va desde 54% a un 98%, siendo este margen muy alto logrando un bajo rendimiento en él. Caso contrario con el descriptor HOG el cual cuenta con una menor dispersión y mayor precisión que va desde un 97% a 99%, dejando al descriptor no solo con buena precisión si no, también con exactitud por estar cerca del valor ideal, el cual es un 100% al momento de clasificarlo.

**Clase "Rocas-Finas-Medianas"** El descriptor HOG obtuvo el mejor rendimiento para este tipo de mineral, con un valor AUC de 99 % de área y un 1 % de varianza. Nuevamente superando al descriptor LBP en un 11 % de precisión global. Mostrando no solo lo preciso si no lo exacto que puede alcanzar al momento de detectar el mineral, ver Figura 5.3.

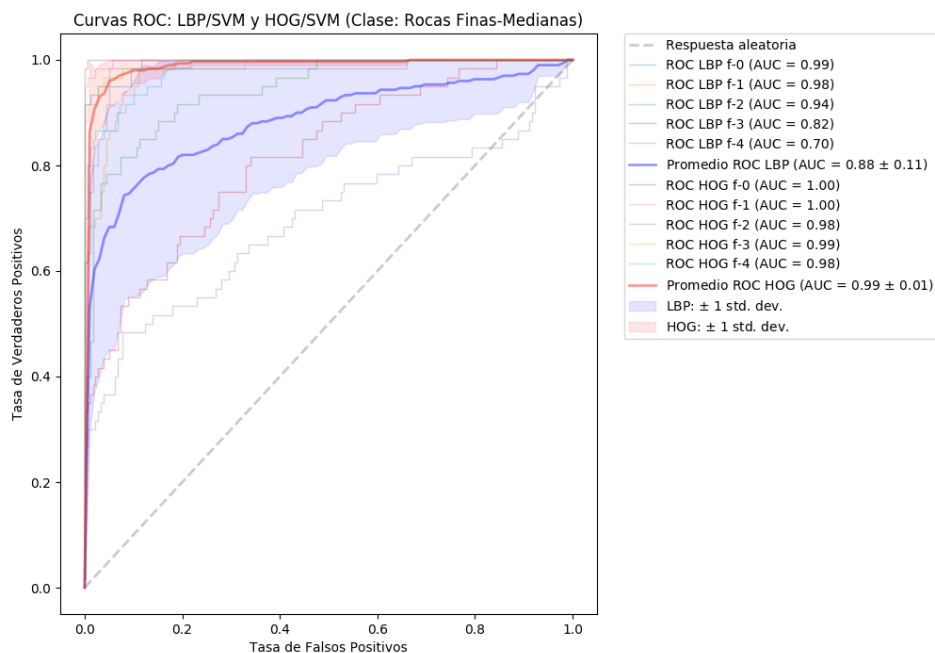


FIGURA 5.3: Curva ROC para la clase "Rocas Finas-Medianas"

**Clase "Rocas-Medianas-Gruesas"** Los dos descriptores vuelven a tener un resultado óptimo en compañía, donde el LBP sobrepasa el HOG por un AUC de 1 % de rendimiento y una varianza 1 % , ver Figura 5.4.

### 5.3.2. Eficiencia computacional

La eficiencia de los dos algoritmos implementados hace énfasis al tiempo en segundos de la velocidad de procesamiento en la ejecución de clasificación del mineral por la cantidad de imágenes.

Como se puede apreciar en la Tabla 5.7 se obtienen dos tiempos distintos que corresponden a Entrenamiento y Pruebas por cada uno de los 5-fold evaluados. Ambos descriptores procesaron la

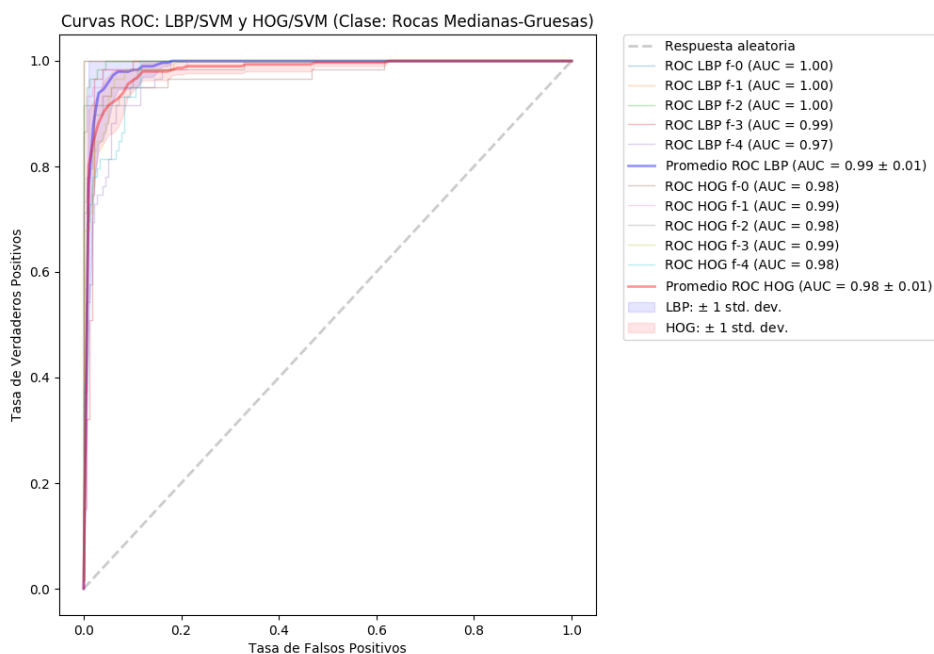


FIGURA 5.4: Curva ROC para la clase "Rocas Medianas-Gruesas".

TABLA 5.7: Tiempo de Ejecución en segundo por cada uno de los descriptores, para la clasificación de mineral.

Validación Cruzada	Tiempo de Ejecución [seg]			
	Entrenamiento		Pruebas	
	LBP	HOG	LBP	HOG
<i>Fold 1</i>	231.29 (957)	112.06 (957)	7.67 (240)	3.83 (240)
<i>Fold 2</i>	232.65 (957)	110.70 (947)	7.88 (240)	3.60 (240)
<i>Fold 3</i>	233.74 (958)	109.89 (958)	7.63 (239)	3.74 (239)
<i>Fold 4</i>	235.43 (958)	106.67 (958)	7.98 (239)	3.58 (239)
<i>Fold 5</i>	233.86 (958)	105.30 (958)	8.20 (239)	3.59 (239)
<b>Promedio</b>	233.39 (957)	108.92 (957)	7.87 (239))	3.66 (239)

misma cantidad de imágenes dejando en igualdad para posteriormente comparar sus resultados.

Se observa el tiempo de entrenamiento promedio de procesar 957 imágenes de los descriptores, donde el LBP obtuvo 233.39 [seg] y el HOG 108.92 [seg], siendo este resultado por debajo de la mitad del LBP; al mismo tiempo para las pruebas el LBP obtuvo 7.87 [seg] siendo aproximadamente el doble del HOG con un valor de 3.66 [seg] en igualdad de condiciones. De esta manera HOG posee la eficiencia óptima al momento de clasificar minerales en procesos continuos.

# Capítulo VI

## Conclusiones y trabajos futuros

### 6.1. Conclusiones

En el trabajo, se investigó el problema de estimar el tamaño de minerales de cobre por medio de imágenes digitales. Se propuso un enfoque de visión por computador basada en la caracterización de la textura de la imagen con minerales de cobre. La implementación y comparación del rendimiento de dos descriptores de textura como el LBP[8] y HOG [38] fueron realizados utilizando un clasificador SVM para las diferentes clases de mineral: Rocas Finas, Rocas Medianas, Rocas Finas-Medianas y Rocas Medianas-Gruesas. Para la primera aproximación se estudió el descriptor LBP por ser implementado para la clasificación de texturas en caras, baldosas, entre otras. El segundo, el descriptor HOG por ser un descriptor relativamente nuevo en los procesos para la clasificación de rostros, objetos, animales entre otros.

En la tesis se realizó un estudio detallado del estado del arte donde se pudieron determinar aquellos métodos y descriptores que podían ser ocupados para la identificación de minerales. Al final de este estudio se concluyó que existen dos métodos de implementación para la identificación y clasificación de minerales: Segmentación y Textura. Donde la segmentación fue el análisis más ocupado en los procesos desde 1980 para estimar el tamaño de minerales, basado en la detección de bordes, operaciones morfológicas y el uso de la transformada *Watershed* [75, 76, 77, 78]. Sin embargo este tipo de clasificación de minerales presenta problemas al momento de sus configuraciones debido a que deben ser ajustados manualmente para cada tipo de situación dejándolo no robusto a los cambios del proceso [79].

Fue por ello que se presentaron nuevos métodos de clasificación de los distintos tipos de minerales ocupando otras características de la imagen como la textura, color y forma. Es allí donde la textura tomo importancia para este tipo de investigaciones. Donde los estudios relevantes encontrados para la clasificación de minerales utilizando descriptores de textura fueron la matriz de co-ocurrencia, Gabor, wavelet, Haar, el algoritmo Varma y Zisserman [99, 100, 101, 102, 103, 104, 105] siendo estos los estudios más relevantes realizados para la clasificación de minerales por medio de la textura. Por otro lado el LBP se ha ocupado para la clasificación de baldosas de granito, logrando una precisión a la hora de clasificación del 94 %, resultado óptimos [104]. Mientras el descriptor HOG no se han ocupado para la naturaleza de esta tesis.

Uno de los primeros problemas encontrados para la implementación del sistema fue la generación de la base de datos, debido a que se debe tener en cuenta los cambios de luminosidad, el enfoque de la cámara, la altura de la cámara al mineral, el tamaño de la imagen a procesar. Buscando un ambiente adecuado para el procesamiento de la imagen. La cantidad homogénea de la base de datos respecto a los distintos tipos de mineral fue fundamental a la hora de la clasificación, para que no se presentaran sesgos en el entrenamiento de las clases de los distintos tipos de mineral, buscando una comparación en igualdad de condiciones. Al igual se concluyó que tener una imagen de gran tamaño de píxeles nos permitía entrenar la imagen con tamaños inferiores a ella para obtener el tamaño ideal del sistema. Debido que al realizar la reducción de la imagen no se pierde información de ella, caso contrario si se desea aumentar de tamaño de una imagen de poco tamaño. Donde los mejores resultados se encontraron en imágenes de  $160 \times 160$  *pixels* para el descriptor LBP y  $1280 \times 1280$  *pixels* para el descriptor HOG.

Los parámetros del descriptor LBP para tener la mayor precisión de su sistema fue con un radio de 4, una cantidad de puntos de 16 y un tamaño de la imagen a procesar de  $160 \times 160$  *pixels*, logrando una precisión del 90,8 % mientras para el HOG fue un tamaño de la imagen de  $1280 \times 1280$  *pixels* con un tamaño de celda de  $32 \times 32$  *pixels* y un bloque de  $1 \times 1$  y una precisión de 96,1 %. Los parámetros para la clasificación con la ayuda del SVM del descriptor LBP fueron con un Gamma de 0,000001 y un C de 10, mientras para el HOG un Gamma de 0,001 y un C de 100. Con estas condiciones se lograron los rendimientos óptimos de los descriptores y clasificador.

Por otro lado se realizaron experimentos con estos parámetros ya ajustados para ver el rendimiento por etiqueta del mineral, donde se observó que las etiquetas más extremas como mineral Fino y Medianas-Gruesas los rendimientos son altos casi en un 100 % para ambos clasificadores, caso contrario con las dos clases faltantes, donde el rendimiento decae, sin embargo el descriptor HOG mantiene un porcentaje de clasificación alto con una tasa no inferior en un AUC de 97 % en comparación con el LBP de AUC 83 %. Adicionalmente se hicieron experimentos para ver

el rendimiento computacional de los descriptores con la curvas ROC, donde se observó que el descriptor LBP tiene un mayor costo computacional al momento de su entrenamiento y prueba, casi al doble que el descriptor HOG. Permitiendo al HOG darle la oportunidad de tener una velocidad mayor si se desea montar en tiempo real en un proceso industrial, demostrando que hasta en tiempo de procesamiento este descriptor sigue mostrando mejor comportamiento.

A partir de lo inferido anteriormente y respondiendo a la hipótesis del trabajo propuesto en la tesis, se puede decir que el descriptor LBP tiene un rendimiento menor al HOG para la clasificación de mineral. Por el contrario, el descriptor HOG se ve como una buena alternativa para este tipo de implementaciones y en los experimentos destacó sobre el LBP. Por otro lado, se puede agregar que a pesar que el descriptor LBP no tuvo el desempeño esperado se puede atribuir a la cantidad y calidad del conjunto de entrenamiento al igual de las características del mineral. En resumen, se ha logrado evaluar un conjunto de descriptores para la clasificación de mineral donde el HOG obtuvo el mejor rendimiento a nivel de eficiencia tanto en precisión como costo computacional. El LBP solo presenta mejor significativa en la clasificación de dos tipos de mineral. Si bien es cierto, la hipótesis se cumple en su totalidad.

## **6.2. Trabajos Futuros**

Con base a los resultados de este trabajo se realizaran pruebas con otras bases de datos encontradas en el estado del arte para ver la robustez del descriptor HOG en otros tipos de minerales con un mayor número de clases, al igual se quiere hacer una extensión de la tesis realizando probabilidades y trozos de la imagen para hacer una estimación global de la clase, con el fin de exigir al descriptor a partir de otras bases de datos y proponer un método nuevo. Adicionalmente se buscará una implementación del sistema en tiempo real en procesos mineros.

# Apéndice A

## Publicaciones

El primer resultado generado en este trabajo fue la aceptación y revisión de los resultados presentados por el descriptor LBP:

- Pupo, A., Bekios, J. (2018). Development of an Artificial Vision System for the Detection of Rocks with Oversize in Conveyor Belts.

# BIBLIOGRAFÍA

- [1] ConcretOline. (2011) [www.concretonline.com/index.php?option=com\\_contentview=articleid=2065catid=35:articulos-tecnicos](http://www.concretonline.com/index.php?option=com_contentview=articleid=2065catid=35:articulos-tecnicos).
- [2] C. A. ZAVALA MEDINA, “Plan de mantenimiento preventivo basado en rcm para el chancador primario fuller, operación mantoverde,” 2018.
- [3] J. B. Pacheco Rubio *et al.*, “Diseño e implementación de un motor de recomendación de tweets,” 2016.
- [4] G. Van Rossum and F. L. Drake Jr, “Tutorial python,” *Fred L. Drake, Jr*, 2009.
- [5] G. Bradski and A. Kaehler, “Opencv,” *Dr. Dobb’s journal of software tools*, vol. 3, 2000.
- [6] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, “scikit-image: image processing in python,” *PeerJ*, vol. 2, p. e453, 2014.
- [7] Scikit. <http://scikit-image.org/docs/dev/api/skimimage.feature.htmlskimimage.feature.hog>.
- [8] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
- [9] S. Chatterjee, A. Bhattacharjee, B. Samanta, and S. K. Pal, “Image-based quality monitoring system of limestone ore grades,” *Comput. Ind.*, vol. 61, no. 5, pp. 391–408, Jun. 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.compind.2009.10.003>
- [10] C. A. Perez, P. A. Estévez, P. A. Vera, L. E. Castillo, C. M. Aravena, D. A. Schulz, and L. E. Medina, “Ore grade estimation by feature selection and voting using boundary detection in digital image analysis,” *International Journal of Mineral Processing*, vol. 101, no. 1, pp. 28 – 36, 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0301751611001037>

- [11] A. K. Patel, S. Chatterjee, and A. K. Gorai, "Development of online machine vision system using support vector regression (svr) algorithm for grade prediction of iron ores," in *Machine Vision Applications (MVA), 2017 Fifteenth IAPR International Conference on*. IEEE, 2017, pp. 149–152.
- [12] F. J. Galdames, C. A. Perez, P. A. Estévez, and M. Adams, "Classification of rock lithology by laser range 3d and color images," *International Journal of Mineral Processing*, vol. 160, pp. 47–57, 2017.
- [13] J. Tessier, C. Duchesne, and G. Bartolacci, "A machine vision approach to on-line estimation of run-of-mine ore composition on conveyor belts," *Minerals Engineering*, vol. 20, no. 12, pp. 1129–1144, 2007.
- [14] C. A. Perez, J. A. Saravia, C. F. Navarro, D. A. Schulz, C. M. Aravena, and F. J. Galdames, "Rock lithological classification using multi-scale gabor features from sub-images, and voting with rock contour information," *International Journal of Mineral Processing*, vol. 144, pp. 56–64, 2015.
- [15] M. Kistner, G. T. Jemwa, and C. Aldrich, "Monitoring of mineral processing systems by using textural image analysis," *Minerals Engineering*, vol. 52, pp. 169–177, 2013.
- [16] A. Amankwah and C. Aldrich, "Estimation of rock particle size distribution using wavelet decomposition and morphological operations," *International Journal of Computer Theory and Engineering*, vol. 5, no. 3, p. 450, 2013.
- [17] D. Mery, "Inspección visual automática," in *Primer Congreso Internacional de Ingeniería Mecatrónica, Lima, Perú*, 2002.
- [18] J. Alba and J. Cid, "Reconocimiento de patrones," *Carlos III, Madrid, Universidad Carlos III, Madrid. Retrived from <http://www.gts.tsc.uvigo.es/pi/Reconocimiento.pdf>*, 2006.
- [19] J. M. Artigas, P. C. Perea, and J. P. i Ramo, *Tecnología del color*. Universitat de valencia, 2002, vol. 58.
- [20] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.
- [21] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," *Proceedings of the Alvey Vision Conference 1988*, pp. 147–151, 1988. [Online]. Available: <http://www.bmva.org/bmvc/1988/avc-88-023.html>

- [22] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” pp. 674–679, 1981.
- [23] G. D. Hager and P. N. Belhumeur, “Efficient region tracking with parametric models of geometry and illumination,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 10, pp. 1025–1039, 1998.
- [24] H.-Y. Shum and R. Szeliski, “Construction of panoramic mosaics with global and local alignment,” *International Journal of Computer Vision*, vol. 36, no. 2, pp. 101–130, 2000. [Online]. Available: <http://research.microsoft.com/apps/pubs/default.aspx?id=75614>
- [25] S. Baker and I. Matthews, “Equivalence and efficiency of image alignment algorithms,” *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, pp. 243–245, 2001.
- [26] T. Yamasaki, “Histogram of Oriented Gradients (HOG),” *The Journal of the Institute of Image Information and Television Engineers*, vol. 64, no. 3, pp. 322–329, 2010. [Online]. Available: <http://ci.nii.ac.jp/naid/10025988610/en/>
- [27] S. G. Mallat, “A Theory for Multiresolution Signal Decomposition: The Wavelet Representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.
- [28] P. Langley, “Selection of Relevant Features in Machine Learning,” in *In Proceedings of the AAAI Fall Symposium on Relevance*, 1994, pp. 140–144.
- [29] R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval*. ACM press New York, 1999, vol. 463.
- [30] J. S. DEL POZO and A. R. GARCÍA, “Detección de matrículas mediante características binarias locales (lbp),” 2009.
- [31] T. Mäenpää, *The local binary pattern approach to texture analysis: extensions and applications*. Oulun yliopisto Oulu, 2003.
- [32] D. Harwood, T. Ojala, M. Pietikäinen, S. Kelmán, and L. Davis, “Texture classification by center-symmetric auto-correlation, using kullback discrimination of distributions,” *Pattern Recognition Letters*, vol. 16, no. 1, pp. 1–10, 1995.
- [33] J. D. Alvarado and J. Fernández, “Análisis de textura en imágenes a escala de grises, utilizando patrones locales binarios (lbp),” *ENGI Revista Electrónica de la Facultad de Ingeniería*, vol. 1, no. 1, 2012.

- [34] O. GARCÍA-OLALLA and E. ALEGRE, “Descripción de textura en imágenes utilizando local binary pattern (lbp).”
- [35] M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen, *Computer vision using local binary patterns*. Springer Science & Business Media, 2011, vol. 40.
- [36] W. Ma, L. Huang, and C. Liu, “Advanced local binary pattern descriptors for crowd estimation,” in *Computational Intelligence and Industrial Application, 2008. PACIIA’08. Pacific-Asia Workshop on*, vol. 2. IEEE, 2008, pp. 958–962.
- [37] LBP-U. <https://github.com/Favi0/Favi0.github.io/wiki/Week2-Deteccion>.
- [38] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.
- [39] A. Albiol, D. Monzo, A. Martin, J. Sastre, and A. Albiol, “Face recognition using hog–ebgm,” *Pattern Recognition Letters*, vol. 29, no. 10, pp. 1537–1543, 2008.
- [40] Y. Pang, Y. Yuan, X. Li, and J. Pan, “Efficient hog human detection,” *Signal Processing*, vol. 91, no. 4, pp. 773–781, 2011.
- [41] J. O. Moll, “Detección de personas en secuencias de vídeo en tiempo real,” 2011.
- [42] F. Arranz Aranda, L. Yin, J. M. Qi López Cámara, and P. J. Martín de la Calle, “Interacción persona-computador basada en el reconocimiento visual de manos,” Ph.D. dissertation, Tesis Universidad Complutense de Madrid, 2011.
- [43] A. E. Osorio, S. A. F. Calvo, and L. G. A. Holguín, “Morphological analysis for automatized visual inspection using reduced hog,” in *Computing Colombian Conference (10CCC), 2015 10th*. IEEE, 2015, pp. 278–285.
- [44] H. López Paredes *et al.*, “Detección y seguimiento de objetos con cámaras en movimiento,” B.S. thesis, 2011.
- [45] R. C. González and R. E. Woods, *Tratamiento digital de imágenes*. Addison-Wesley New York, 1996, vol. 3.
- [46] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [47] C. T. Jianbo Shi, “Good Features to Track.pdf,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 593 – 600, 1994.

- [48] H. P. Moravec, "Visual Mapping by a Robot Rover," in *Proceedings of the 6th International Joint Conference on Artificial Intelligence - Volume 1*, ser. IJCAI'79. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1979, pp. 598–600. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1624861.1624997>
- [49] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997. [Online]. Available: [http://ieeexplore.ieee.org/xpls/abs/\\_all.jsp?arnumber=598236](http://ieeexplore.ieee.org/xpls/abs/_all.jsp?arnumber=598236)
- [50] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on pattern analysis and machine*, vol. 24, no. 5, pp. 603–619, 2002. [Online]. Available: <http://doi.ieeecomputersociety.org/10.1109/34.1000236>
- [51] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: theory and its application to image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 11, pp. 1101–1113, 1993. [Online]. Available: <http://dx.doi.org/10.1109/34.244673>
- [52] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation Normalized Cuts and Image Segmentation Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000. [Online]. Available: [http://repository.upenn.edu/cis/\\_papers/\\_5Cnhttp://dx.doi.org/10.1109/34.868688/\\_5Cnhttp://repository.upenn.edu/cis/\\_papers/107](http://repository.upenn.edu/cis/_papers/_5Cnhttp://dx.doi.org/10.1109/34.868688/_5Cnhttp://repository.upenn.edu/cis/_papers/107)
- [53] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.
- [54] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic Active Contours," *International Journal of Computer Vision*, vol. 22, no. 1, pp. 61–79, 1997.
- [55] C.-B. Ng, Y.-H. Tay, and B.-M. Goi, "A review of facial gender recognition," *Pattern Analysis and Applications*, vol. 18, no. 4, pp. 739–755, 2015. [Online]. Available: <http://link.springer.com/10.1007/s10044-015-0499-6>
- [56] H. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, 1998. [Online]. Available: <http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=655647>
- [57] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 153–161, 2005.

- [58] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, "Random Forests for Real Time 3D Face Analysis," *International Journal of Computer Vision*, vol. 101, no. 3, pp. 437–458, 2013.
- [59] C. P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," pp. 555–562, 1998.
- [60] L. M. Mingo Morocho, "Construcción de un prototipo para el reconocimiento y traducción del lenguaje de señas a texto utilizando el sensor kinect." B.S. thesis, 2016.
- [61] V. Riobó Otero, "Reconocimiento de localizaciones mediante máquinas de soporte vectorial," Master's thesis, 2012.
- [62] X. Zhang, C. Jiang, and M.-j. Luo, "Training classifiers for unbalanced distribution and cost-sensitive domains with roc analysis," in *Pacific Rim Knowledge Acquisition Workshop*. Springer, 2006, pp. 89–98.
- [63] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Ijcai*, vol. 14, no. 2. Montreal, Canada, 1995, pp. 1137–1145.
- [64] S. V. Stehman, "Selecting and interpreting measures of thematic classification accuracy," *Remote sensing of Environment*, vol. 62, no. 1, pp. 77–89, 1997.
- [65] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," 2011.
- [66] J. A. Nevin, "Signal detection theory and operant behavior: A review of david m. green and john a. swets'signal detection theory and psychophysics. 1," *Journal of the Experimental Analysis of Behavior*, vol. 12, no. 3, pp. 475–480, 1969.
- [67] L. B. Lusted, "Signal detectability and medical decision-making," *Science*, vol. 171, no. 3977, pp. 1217–1219, 1971.
- [68] M. H. Zweig and G. Campbell, "Receiver-operating characteristic (roc) plots: a fundamental evaluation tool in clinical medicine." *Clinical chemistry*, vol. 39, no. 4, pp. 561–577, 1993.
- [69] R. Smith-Bindman, K. Kerlikowske, V. A. Feldstein, L. Subak, J. Scheidler, M. Segal, R. Brand, and D. Grady, "Endovaginal ultrasound to exclude endometrial cancer and other endometrial abnormalities," *Jama*, vol. 280, no. 17, pp. 1510–1517, 1998.
- [70] J. Piatt, "Receiver-operating characteristic curves." *Journal of neurosurgery*, vol. 95, no. 5, pp. 918–919, 2001.

- [71] H. El-Halwagy, T. Gelbaya, M. El-Wahab, S. El-Din, A. Shebl, and G. Makboul, "The mean third trimester postprandial blood glucose of diabetic pregnant patients and infant birth weight in the kuwaiti population." *Medscape women's health*, vol. 6, no. 5, pp. 2–2, 2001.
- [72] D. G. Altman and J. M. Bland, "Diagnostic tests. 1: Sensitivity and specificity." *BMJ: British Medical Journal*, vol. 308, no. 6943, p. 1552, 1994.
- [73] J. Kaartinen and A. Tolonen, "Utilizing 3d height measurement in particle size analysis," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 3292 – 3297, 2008, 17th IFAC World Congress. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1474667016394587>
- [74] G. Hunter, C. McDermott, N. Miles, A. Singh, and M. Scoble, "A review of image analysis techniques for measuring blast fragmentation," *Mining Science and Technology*, vol. 11, no. 1, pp. 19 – 36, 1990. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/016790319080003Y>
- [75] T. Al-Rousan, E. Masad, E. Tutumluer, and T. Pan, "Evaluation of image analysis techniques for quantifying aggregate shape characteristics," *Construction and Building Materials*, vol. 21, no. 5, pp. 978 – 990, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0950061806000377>
- [76] B. J. L. D. T. E. Masad et al. Masad E, Al RT, *Test Methods for Characterizing Aggregate Shape, Texture, and Angularity*, 2007. [Online]. Available: <https://books.google.cl/books?id=KqyunQAACAAJ>
- [77] R. King, "Determination of the distribution of size of irregularly shaped particles from measurements on sections or projected areas," *Powder Technology*, vol. 32, no. 1, pp. 87 – 100, 1982. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0032591082850092>
- [78] W. Petruk, "Automatic image analysis for mineral beneficiation," *JOM*, vol. 40, no. 4, pp. 29–31, Apr 1988. [Online]. Available: <https://doi.org/10.1007/BF03259018>
- [79] F. Meyer and S. Beucher, "Morphological segmentation," *Journal of visual communication and image representation*, vol. 1, no. 1, pp. 21–46, 1990.
- [80] X. Wu and J. M. Kemeny, "A segmentation method for multi-connected particle delineation," pp. 240–247, Nov 1992.

- [81] N. H. Maerz, "Image sampling techniques and requirements for automated image analysis of rock fragmentation," *Proceedings of the FRAGBLAST*, vol. 5, pp. 115–120, 1996.
- [82] S. Mkwelo, G. De Jager, and F. Nicolls, "Watershed-based segmentation of rock scenes and proximity-based classification of watershed regions under uncontrolled lighting conditions," in *Proceedings of 14th Annual Symposium of the Pattern Recognition Association of South Africa (PRASA)*, 2003, pp. 107–112.
- [83] S. Al-Thyabat and N. Miles, "An improved estimation of size distribution from particle profile measurements," *Powder Technology*, vol. 166, no. 3, pp. 152–160, 2006.
- [84] O. Guyot, T. Monredon, D. LaRosa, and A. Broussaud, "Visiorock, an integrated vision technology for advanced control of comminution circuits," *Minerals Engineering*, vol. 17, no. 11, pp. 1227–1235, 2004.
- [85] J. Kemeny, E. Mofya, R. Kaunda, and P. Lever, "Improvements in blast fragmentation models using digital image processing," *Fragblast*, vol. 6, no. 3-4, pp. 311–320, 2002.
- [86] M. J. Thurley and K. C. Ng, "Identifying, visualizing, and comparing regions in irregularly spaced 3d surface data," *Computer vision and image understanding*, vol. 98, no. 2, pp. 239–270, 2005.
- [87] T. Andersson and M. J. Thurley, "Minimizing profile error when estimating the sieve-size distribution of iron ore pellets using ordinal logistic regression," *Powder technology*, vol. 206, no. 3, pp. 218–226, 2011.
- [88] T. Andersson, M. J. Thurley, and J. E. Carlson, "A machine vision system for estimation of size distributions by weight of limestone particles," *Minerals Engineering*, vol. 25, no. 1, pp. 38–46, 2012.
- [89] S. Tafesse, "Physical characterization of coarse clasts with 3d image-analysis method: development, evaluation and application," Ph.D. dissertation, KTH Royal Institute of Technology, 2012.
- [90] M. J. Thurley and K. C. Ng, "Identification and sizing of the entirely visible rocks from a 3d surface data segmentation of laboratory rock piles," *Computer Vision and Image Understanding*, vol. 111, no. 2, pp. 170–178, 2008.
- [91] M. Thurley, "Measuring the visible particles for automated online particle size distribution estimation," in *International Mineral Processing Congress: 20/10/2014-24/10/2014*. Gecamin, 2014.

- [92] C. Liao and Y. Tarng, "On-line automatic optical inspection system for coarse particle size distribution," *Powder Technology*, vol. 189, no. 3, pp. 508–513, 2009.
- [93] Z. Zhang, X. Su, L. Ding, Y. Wang *et al.*, "Multi-scale image segmentation of coal piles on a belt based on the hessian matrix," *Particuology*, vol. 11, no. 5, pp. 549–555, 2013.
- [94] E. Hamzeloo, M. Massinaei, and N. Mehrshad, "Estimation of particle size distribution on an industrial conveyor belt using image analysis and neural networks," *Powder Technology*, vol. 261, pp. 185–190, 2014.
- [95] Z. Zhang, "Particle overlapping error correction for coal size distribution estimation by image analysis," *International Journal of Mineral Processing*, vol. 155, pp. 136–139, 2016.
- [96] J. Yang and S. Chen, "An online detection system for aggregate sizes and shapes based on digital image processing," *Mineralogy and Petrology*, vol. 111, no. 1, pp. 135–144, 2017.
- [97] R. M. Haralick, K. Shanmugam *et al.*, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.
- [98] B. Julesz *et al.*, "Textons, the elements of texture perception, and their interactions," *Nature*, vol. 290, no. 5802, pp. 91–97, 1981.
- [99] D. Moolman, C. Aldrich, J. Van Deventer, and W. Stange, "The classification of froth structures in a copper flotation plant by means of a neural net," *International Journal of Mineral Processing*, vol. 43, no. 3-4, pp. 193–208, 1995.
- [100] J. Kaartinen, J. Hätönen, H. Hyötyniemi, and J. Miettunen, "Machine-vision-based control of zinc flotation—a case study," *Control Engineering Practice*, vol. 14, no. 12, pp. 1455–1466, 2006.
- [101] G. Barbery and D. Leroux, "Prediction of particle composition distribution after fragmentation of heterogeneous materials," *International Journal of mineral processing*, vol. 22, no. 1-4, pp. 9–24, 1988.
- [102] K. Petersen, C. Aldrich, and J. Van Deventer, "Analysis of ore particles based on textural pattern recognition," *Minerals Engineering*, vol. 11, no. 10, pp. 959–977, 1998.
- [103] G. T. Jemwa and C. Aldrich, "Estimating size fraction categories of coal particles on conveyor belts using image texture modeling methods," *Expert Systems with Applications*, vol. 39, no. 9, pp. 7947–7960, 2012.

- [104] F. Bianconi, E. González, A. Fernández, and S. A. Saetta, “Automatic classification of granite tiles through colour and texture features,” *Expert Systems with Applications*, vol. 39, no. 12, pp. 11 212–11 218, 2012.
- [105] G. Bull, J. Gao, and M. Antolovich, “Rock fragment boundary detection using compressed random features,” in *International Joint Conference on Computer Vision, Imaging and Computer Graphics*. Springer, 2014, pp. 273–286.
- [106] Z. Zhang and J. Yang, “Online analysis of coal ash content on a moving conveyor belt by machine vision,” *International Journal of Coal Preparation and Utilization*, vol. 37, no. 2, pp. 100–111, 2017.
- [107] B. H. Kumar *et al.*, “Recognition of human actions in videos using computer vision techniques,” *International Journal of Innovations in Engineering and Technology (IJIET)*, vol. 7, no. 3, 2016.
- [108] V. Tripathi, A. Mittal, D. Gangodkar, and V. Kanth, “Real time security framework for detecting abnormal events at atm installations,” *Journal of Real-Time Image Processing*, pp. 1–11, 2016.
- [109] J. Bekios. (2017) mail. [Online]. Available: [juan.bekios@ucn.c](mailto:juan.bekios@ucn.c)