

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación

**Clasificación de imágenes de obras de arte en su estilo pictórico
considerando características de bajo nivel**

Tesis Presentada como Requisito para Obtener el Título de:
Licenciatura en Ingeniería en Ciencias de la Computación

Presenta
Jorge Navarro González

Asesor
Dra. María de la Concepción Pérez de Celis Herrero

H. Puebla de Zaragoza

Resumen

En este trabajo se presenta un estudio sobre las características de bajo nivel que pudieran servir para construir un sistema automático no supervisado tendiente a clasificar obras de arte (en particular pinturas de caballete) de acuerdo a su estilo pictórico. Se analizan en particular tres estilos: Impresionista, Barroco y Cubista y utilizando MatLab se desarrollaron una serie de algoritmos que permite extraer de las imágenes de las piezas de los estilos mencionados, sus características de bajo nivel en particular se extraen descriptores de oscuridad, iluminosidad, así como textura. Posteriormente se utilizan técnicas de Minería de Datos para el procesamiento de las características y la clasificación de las mismas.

INDICE

1.	Introducción	4
1.1.	Contexto	7
1.2.	Objetivos.....	7
2.	Fundamentos Teóricos.....	8
2.1.	Modelos de color.....	8
2.1.1.	Modelo RGB	8
2.2.	Modelo HSV	9
2.3.	Textura	10
2.3.1.	Modelos Estructurales	10
2.3.2.	Modelos Estadísticos	10
2.3.3.	Modelos Espectrales	11
2.4.	Histograma de color	12
2.5.	Medidas de Similitud	12
2.6.	Algoritmos de Clasificación.....	13
2.6.1.	Algoritmos de clasificación supervisados	13
2.6.2.	Ejemplos de algoritmos de clasificación supervisados	13
2.6.2.1.	Algoritmo C4.5	13
2.6.2.2.	Algoritmo ID-3 (Induction Decision Trees)	14
2.6.2.3.	Redes neuronales	15
2.6.3.	Algoritmos de clasificación no supervisados	15
2.6.3.1.	Algoritmos de agrupamiento (<i>clustering</i>).....	15
2.6.3.1.1.	Modelos de conexión	16
2.6.3.1.2.	Modelos de centroides	16
2.6.3.1.3.	Modelos de distribución	16
2.6.3.1.4.	Modelos de densidad	17
2.6.4.	Ejemplos de Algoritmos de clasificación no supervisados ..	17
2.6.4.1.	K-means.....	17
2.6.4.2.	Algoritmo de esperanza-maximización (Algoritmo EM) ..	17
2.6.4.3.	DBSCAN	18
2.7.	Metodología.....	18
3.	Propuesta de solución.....	20
3.1.	Objetivos Específicos.....	20
3.2.	Construcción de la base de datos de imágenes de obras pictóricas.....	23
3.3.	Características de iluminación	29
3.4.	Características de textura	30
3.5.	Características de los bordes	31
4.	Resultados obtenidos.....	31
4.1.	Resultados obtenidos utilizando el algoritmo C4.5	35
5.	Conclusiones y trabajos a futuro.....	41
6.	Bibliografía	42

1. Introducción

El análisis computacional de imágenes nace por la búsqueda de métodos para detectar ciertas características en radiografías o tomografías para la detección de patrones que podrían significar algún riesgo para la salud. Estos sistemas demostraron que, además de poder realizar esa función, eran capaces de realizar otras tareas, como clasificar distintos tipos de imágenes. Las técnicas utilizadas en otros campos de manera natural se han extendido y utilizado también en el campo de arte, ya que la imagen es su elemento fundamental.

A raíz del uso de Sistemas de Recuperación de Imágenes en Arte, surgen de manera natural las preguntas de investigación siguientes: ¿Se puede emular a un experto y clasificar de manera automática una obra de arte por su estilo pictórico?, ¿Se pueden utilizar, características de bajo nivel para la extracción de contenidos en las imágenes y determinar a partir de ellos su estilo pictórico?, ¿Qué características son las que se deben considerar par la tarea de clasificación por estilo pictórico: La paleta de color empleada, el uso de la luz, la línea, la composición, la profundidad o una composición de todas ellas?

Teniendo en perspectiva las preguntas consideradas en el párrafo precedente, en este trabajo de tesis nuestro objetivo es, desarrollar e implementar un sistema que pueda clasificar pinturas de caballete, en sus respectivos estilos artísticos (o pictóricos). Como ejemplo de lo que esperamos, de un sistema de este tipo es, que al dar la imagen de la pintura de Claude Oscar Monet *En el acantilado*(1873), mostrada en la figura 1. el sistema sea capaz de clasificarla como Impresionista.



Figura 1. Imagen de la pintura impresionista *En el acantilado*, de Claude Oscar Monet. El sistema a desarrollar deberá poder extraer *ciertas* características de la imagen y compararla con las características de los diferentes estilos pictóricos para poder determinar el estilo al que pertenece la imagen objetivo.

Resulta claro que la intención con esta investigación es por una parte determinar qué características(descriptores) de bajo nivel pueden utilizarse para clasificar las imágenes de las obras pictóricas por su estilo ya que los expertos utilizan estas características y otras de carácter visual e iconográfico para realizar su diagnóstico. Una vez determinadas estas características utilizando diferentes algoritmos de clasificación ver si las características encontradas son suficientes para realizar el diagnóstico esperado y cuál de los algoritmos de clasificación es el más adecuado para esta tarea.

Cuando nos referimos a utilizar descriptores de bajo nivel consideramos: luminosidad, textura y color entre otras. Para la extracción de estas características se pretende desarrollar una serie de algoritmos que permitan la recuperación de dichos descriptores. Las características consideradas posteriormente deben ser analizadas para determinar su importancia en el momento de la clasificación. Una de las preguntas asociadas a la utilidad de las características seleccionadas sería ¿El índice de oscuridad en la imagen es determinante para distinguir entre impresionismo y barroco?. Existen en la historia de la pintura numerosos estilos pictóricos o ismos por lo que este trabajo se reduce a considerar solamente una selección de éstos. Las imágenes consideradas como caso de estudio serán obras de arte de los estilos barroco, impresionismo y cubista. Los usos posibles para un sistema de este tipo son diversos: El sistema podría ser utilizado como una herramienta educativa en la enseñanza de la historia del arte; Análisis de arte de su uso personal o bien; Se podría tener una aplicación móvil que permita que el dispositivo capture una fotografía de una pintura en un museo, la analice y proporcione al usuario información sobre la pintura

En un sistema de recuperación de imagen de arte por lo general se tienen al menos tres componentes: análisis formal, comparación de los aspectos formales de pinturas, y la clasificación de estilo. Existen varios modelos documentados de búsqueda para imágenes de arte, algunos de ellos [1][2] son:

- **Búsqueda por Asociación.** Para este modelo se debe analizar todas las imágenes contenidas en la base de datos, considerando en este análisis los criterios dados de semejanza para una imagen dada o el bosquejo. Esto a menudo implica el refinamiento iterativo de la búsqueda.

- **Búsqueda Dirigida.** El objetivo en este tipo de búsqueda es el de encontrar una imagen específica. La búsqueda, puede ser para una copia exacta de la imagen dada como ejemplo o la búsqueda objetivo también puede ser para otra imagen del mismo objeto es decir supongamos que el usuario tiene una imagen determinada y desea encontrar imágenes con los mismos objetos por ejemplo otra pintura de la serie Marilyn de Andy Warhol (1962), un ejemplo de este tipo de búsqueda se muestra en la Figura 2. Para las búsquedas dirigidas debe especificarse el tipo de imágenes objetivo es decir: imagen exacta o bien mismo objeto con características diferentes desde los aspectos de cromaticidad, textura, etc.



Figura 2. Ejemplo de búsqueda dirigida, en este caso se buscan imágenes del mismo objeto con características cromáticas diferentes.

- **Búsqueda por Categoría.** El objetivo es encontrar a un representante arbitrario que corresponda a la clase específica de la imagen muestra. En la

búsqueda por categoría, el usuario puede tener disponible un grupo de imágenes y la búsqueda es para las imágenes adicionales de la misma clase. Por ejemplo, buscar otras pinturas del mismo artista o del mismo estilo pictórico.

Trabajando con imágenes de arte, podemos identificar también, dos tipos intervalos en la diversidad de significados: el intervalo sensorial (*sensory gap*) y el intervalo de contenido o semántico (*semantic gap*). El Intervalo Sensorial se relaciona con la diversidad entre el objeto en el mundo real y el que se observa en la pintura analizada [3]. Para el análisis digital de las imágenes se tiene entonces que considerar el uso del color, la existencia de bordes, la luminosidad y la textura considerando para ello las propiedades estadísticas de las imágenes. Para el caso, del intervalo de contenido o semántico hace referencia a diferencia entre la interpretación inmediata de información pictórica en todas sus formas diferentes y la interpretación que sigue de una descripción formal del objeto [4].

Para poder indicar formalmente un estilo pictórico el observador debe entender el arte en los términos formales del color, la forma (lineal o pictórica) y la iluminación además del contenido iconográfico [5] [6]. Cada uno, de los elementos formales por razones de claridad, en la explicación de nuestra técnica diremos que corresponden a una categoría. Tendremos de esta forma cuatro categorías a analizar: Luz, Línea, Textura y Color. No consideraremos en estas categorías el análisis iconográfico, ya que para ello se requiere la identificación y connotación de los objetos mediante una interpretación semántica que queda fuera de los alcances de este trabajo.

El uso, de la Luz en una pintura se relaciona estrechamente con el color y la composición, la degradación de la luz contribuye a generar atmósferas y a producir efectos de profundidad, el uso de la luz es por sí misma una característica relevante cuando de clasificación de estilos pictóricos se trata, como es en el caso del impresionismo donde la luz es la protagonista. Cuando en pintura se hace referencia a la Línea se habla de composición y organización espacial pero también de trazo, es decir si las formas que constituyen las imágenes creadas por el pintor son producidas por dibujo o creadas por veladuras como es el caso de la pintura renacentista con el uso de la técnica del *sfumato*. La Textura define la calidad de la superficie en términos de su apariencia, la textura real hace referencia a la característica física de la pincelada por ejemplo, el *impasto* es la aplicación del pigmento de color de forma poco fluida creando una capa densa, una pintura tiene también una textura virtual lograda por los efectos realizados por el pintor que pueden darle apariencia rugosa siendo esta diferente a su textura real. Finalmente el Color es quizá el elemento formal más conocido, cuando se habla de color consideramos tres elementos: el tinte, la saturación y el brillo o intensidad. En pintura es común hablar de la paleta de color de determinada época o estilo pictórico siendo ésta la gama de colores utilizada en las obras de las épocas correspondientes, por ejemplo para el caso de Picaso y Braque pintores cubistas su paleta tiene de colores apagados (ocres, verdes oscuros y diversos grises).

Consideramos principalmente en esta etapa del proyecto, tres estilos de arte: Barroco, Cubismo e Impresionismo.

El núcleo del trabajo realizado para alcanzar el objetivo propuesto fue, entonces, el análisis de las características relevantes que consideran los especialistas en historia del arte, esto nos permitió desarrollar es un conjunto de algoritmos para extracción de estas características mediante el análisis de las imágenes pertenecientes a los estilos pictóricos ya mencionados. Una vez extraídas las características, se requiere de la utilización de algoritmos de aprendizaje supervisado para llevar a cabo la categorización de estilos.

Cuando nos referimos, a algoritmos de extracción de características, queremos decir que estos algoritmos obtienen las características de bajo nivel de la imagen las cuales pueden interpretarse como características pictóricas cuantificables. Por ejemplo, una característica relevante en las pinturas impresionistas es el manejo de la luz y la carencia de colores oscuros, este hecho se puede cuantificar estudiando el histograma de grises y obteniendo el porcentaje de píxeles oscuros de la imagen. Otro ejemplo es el algoritmo de recuento de borde el cual podría informar cuántos bordes definidos están presentes en la imagen (línea). Estos datos cuantitativos de la pintura se organizan y con ellos se alimenta un algoritmo de aprendizaje. El algoritmo de clasificación, utiliza esta información para aprender a discriminar entre los estilos de arte, por ejemplo, podría notar que las pinturas con una amplia variedad de colores tienden a ser expresionista abstracta. En este proyecto se consideraron los algoritmos de clasificación *k-means* y árboles de decisión

1.1. Contexto

Actualmente un número importante de museos y galerías han incorporado, en la web, como servicio al público la visualización de sus colecciones. Una de las ventajas de contar con este tipo de herramientas, es tener la posibilidad de que el público visitante o los estudiosos de la historia del arte, puedan tener acceso a visitar la colección considerando sus intereses personales. Para agilizar esta clasificación resulta útil el poseer algún método de clasificación que, de forma automática pueda separar las obras de arte según ciertas características como es su estilo artístico.

Esto nos plantea dos problemas, la extracción de las características y la correcta clasificación de las obras de arte. En esta tesis se comparan distintos métodos de clasificación y se ponen a prueba algunos algoritmos para comparar su eficiencia para solucionar este problema.

1.2. Objetivos

Esta tesis sigue la línea de la recuperación de información en imágenes mediante sistemas basados en contenido. En concreto se analizan los descriptores de bajo nivel de distintas obras para construir un vector característico, el cual se utilizara para clasificar dichas obras de arte y finalmente comparar los resultados de dichas clasificaciones para evaluar sus resultados.

2. Fundamentos Teóricos

En este capítulo se describirán todos aquellos aspectos teóricos importantes para tener un buen entendimiento de los conceptos utilizados para realizar la implementación del sistema.

2.1. Modelos de color

Un modelo de color¹ describe matemáticamente cómo pueden ser representados los colores. Estos colores (modelos) pueden clasificarse en los que están orientados a los dispositivos y los que están orientados al tratamiento por parte del usuario. Los primeros hacen especial mención en cómo se registra la información de color de la luz en un dispositivo como una cámara CCD (RGB) o que tratamiento se hace por parte de los dispositivos de impresión para obtener el color (espacio CMY-CMYK, YIQ). Los segundos hacen referencia a cómo el usuario busca una mejor representación para facilitar tareas de procesamiento de la información registrada en la imagen (HSV y sus variantes, CIE y sus variantes). Mientras que el espacio RGB tiene una distribución lineal, los espacios de procesamiento como el HSV y sus variantes tienen una distribución no-lineal de sus tres componentes [7].

Entre los espacios de color mencionados anteriormente, solo se van a destacar dos de ellos, RGB y HSV. El espacio lineal RGB por ser el espacio de color que utilizan la gran mayoría de cámaras de video y fotográficas para construir una imagen de color. Por otro lado, el espacio no lineal HSV, representa uno de los espacios de coordenadas más clásicos e intuitivos existentes en la literatura, capaz de obtener una representación del color a partir de las tres características de la luz que definen el color.

2.1.1. Modelo RGB

El espacio RGB se representa como un cubo donde un color está definido por la mezcla de valores de intensidad de tres colores primarios, rojo, verde y azul (ver Figura 3). Un color viene descrito por una tupla de 3 coordenadas en el cubo. El color negro se representa por ($r=0$, $g=0$, $b=0$) y el color blanco por ($r=255$, $g=255$, $b=255$). La gama acromática de escala de grises está representada por la diagonal del cubo [8].

¹ Un modelo de color es la especificación de un sistema de coordenadas tridimensional y de un sub-espacio de éste sistema en el que cada color es representado por un único punto.

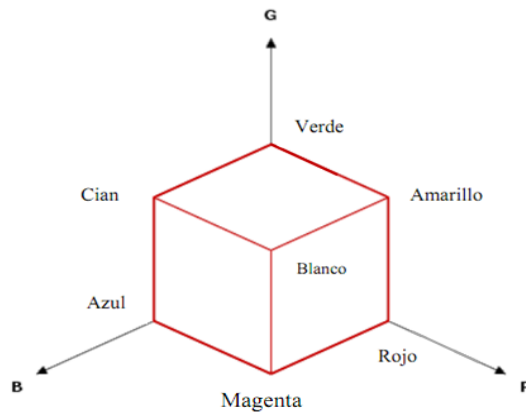


Figura 3. Representación del espacio RGB.

2.2. Modelo HSV

En los últimos años, los espacios de color basados en coordenadas polares se han extendido considerablemente en los sistemas de procesamiento de imagen y detección. Estos espacios de color presentan ciertas ventajas frente al RGB, ya que se asemejan al modo en que el ser humano intuye los colores y es capaz de separar valores cromáticos de los que no lo son. Su interpretación geométrica viene determinada por un cono de base quasi-hexagonal Smith [9]. Con esta representación del espacio de color, cada color trabaja con 3 componentes básicas: matiz, saturación y brillo. El matiz H , hace referencia al valor de cromaticidad o clase de color. La saturación, S , se refiere a las longitudes de onda que se suman a la frecuencia del color, y determina la cantidad de blanco que contiene un color. Cuando menos saturado esté un color más cantidad de blanco y cuanto más saturado esté un color menor cantidad de blanco. En definitiva, la saturación representa la pureza e intensidad de un color. Así, la falta de saturación viene dada por la generatriz en la representación del cono HSV (ver Figura 4). Esa falta de saturación representa la gama de grises desde el blanco hasta el negro. La luminancia V , se corresponde con la apreciación subjetiva de claridad y oscuridad. Cuando se requiere representar una imagen en color con un espacio de color HSV, es importante determinar cómo influyen las componentes de color de una cámara de video RGB sobre el espacio HSV [7].

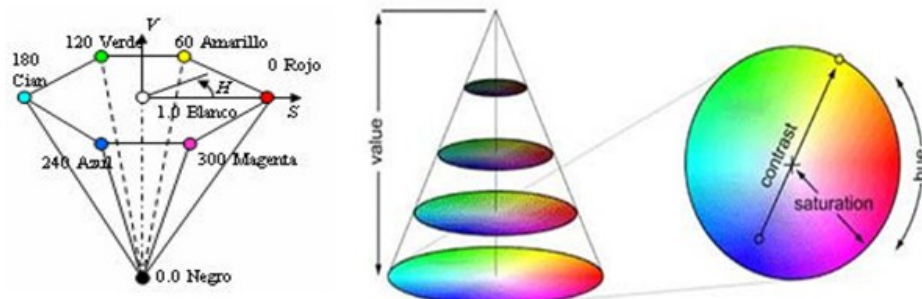


Figura 4. Representación del espacio HSV.

Existen otros modelos de color como XYZ, CIELAB, CIELUV, YUV y HSI, los cuales se describen en [10].

2.3. Textura

Para caracterizar la textura de una superficie de una imagen, se toma en cuenta la distribución de valores de intensidad del color a nivel espacial, permitiendo evaluar propiedades como: fineza, rugosidad, suavidad, granulación, aleatoriedad, etc. En una textura formada por la repetición periódica de un patrón, la repetición es regular o estructurada, como por ejemplo dibujos en una tela y una pared de ladrillos, que se presentan en los ejemplos a) y b) de la Figura 5. En el caso de parcialmente periódica, el patrón se presenta de manera irregular o aleatoria, como por ejemplo tejidos de intestinos humanos, como los ejemplos c) y d) de la Figura 5 [11].



Figura 5. Tipos de texturas

La textura en una imagen depende mucho de la resolución de la misma. Lo que en una escala de resolución son objetos claramente distinguibles, a una escala de resolución menor se puede observar cierta textura y a una escala de resolución todavía menor puede parecer una región uniforme.

El analizar y reconocer diferentes tipos de textura es útil para el reconocimiento de ciertas clases de objetos e incluso en otros aspectos de visión como la determinación de forma tridimensional (*shape from texture*). Existen diferentes formas de describir los tipos de textura, y se clasifican de la siguiente manera:

2.3.1. Modelos Estructurales

Los métodos estructurales definen las texturas como una composición de elementos primitivos bien definidos, por ejemplo, líneas paralelas regularmente espaciadas [12], concibiendo las texturas reales o naturales, como una distorsión de estas texturas ideales. Otros enfoques pueden considerar a las texturas como una colección de objetos primitivos similares (pero no idénticos) distribuidos con algún patrón de repetición. La caracterización de las texturas bajo este enfoque puede realizarse calculando propiedades estadísticas sobre esta distribución de elementos, o las reglas de distorsión y/o ubicación [13].

2.3.2. Modelos Estadísticos

El modelo estadístico [14] es considerado el más importante que se ha propuesto para el análisis de textura. Este modelo utiliza los rasgos

estadísticos de la textura de la imagen, en la cual la textura es modelada como una función de los tonos de gris. Es obvio que el sistema visual humano es sensible para diferenciar muchos rasgos estadísticos de la textura. Este hecho es exitosamente utilizado para hacer la distinción entre diferentes regiones de textura. Desde el punto de vista estadístico se observa que una textura puede definirse por un conjunto de datos estadísticos extraídos de un gran conjunto de propiedades locales de la imagen. Según Haralick 1979 [15], la textura de una imagen de niveles de gris está caracterizada por dos dimensiones. La primera dimensión comprende los parámetros de la distribución de los valores de gris que generan un procedimiento estadístico simple llamado de “primer orden”, donde están incluidos los momentos estadísticos correspondientes al histograma de los niveles de gris de la región y la entropía media. Estos rasgos de textura poseen la limitación que no contienen información respecto a la posición relativa que tienen los píxeles unos con respecto a los otros. Una vía para obtener este tipo de información, en el proceso del análisis de textura es considerar no solamente la distribución de las intensidades, sino también las posiciones de los píxeles con iguales valores de intensidad o cercanamente igual. Para ello fue diseñada la segunda dimensión que comprende las relaciones y dependencias que existen entre los niveles de gris vecinos o aún más, entre regiones de niveles de gris de la imagen lo cual genera otro procedimiento estadístico llamado de “segundo orden”. Esta segunda dimensión es la base primaria para el desarrollo de los métodos de análisis de textura que se basan en la matriz de co-ocurrencia de los niveles de gris (Haralick et al., 1973 [16]), la matriz run length (Galloway, 1975 [17]), la suma y la diferencia de los niveles de gris.

2.3.3. Modelos Espectrales

La transformada de Fourier es adecuada en describir información “global” en la imagen, en especial patrones periódicos. Este es el caso de las texturas, generalmente, por lo que los modelos espectrales proveen buenas características para su descripción y clasificación. En particular, hay 3 características del espectro que son adecuadas para la descripción de texturas [18]:

1. La amplitud de los picos prominentes dan la dirección principal de los patrones en la textura.
2. La localización de los picos en frecuencia indican el periodo espacial de los patrones.
3. Eliminando componentes periódicas mediante filtros en Fourier, se pueden dejar solo las componentes periódicas a las que se les aplica técnicas estadísticas.

En [14] se presenta un análisis detallado de texturas.

2.4. Histograma de color

El histograma de una imagen está definido por la función de masa de probabilidad de las intensidades de la imagen. Esto se extiende para imágenes en color para capturar las probabilidades conjuntas de las intensidades de los tres canales de color. Formalmente, el histograma de color se define por,

$$H_{A,B,C}(a,b,c) = N * \Pr(A = a, B = b, C = c) \quad (1)$$

Donde A, B y C representan los tres canales de color (R, G, B o H, S, V) y N, es el número de píxeles en la imagen. Computacionalmente, el histograma de color está formado por la discretización de los colores de una imagen y el número de píxeles de cada color [20]. Para generar los histogramas de color se necesita, un espacio de color² el cual describe matemáticamente cómo pueden ser representados los colores. De los espacios de color existentes solo se trabajó con dos de ellos RGB por ser el espacio de color más extendido y el que utiliza la gran mayoría de cámaras de video y fotográficas y, HSV por que se asemejan al modo, en que el ser humano intuye los colores y es capaz de separar valores cromáticos de los que no lo son.

2.5. Medidas de Similitud

Las medidas de similitud son utilizadas para establecer una correspondencia adecuada entre imágenes, algunas medidas son:

■ Distancia Euclidiana:

$$\sum_{a=0}^{N-1} \sum_{b=0}^{N-1} \sum_{c=0}^{N-1} (H_1(a,b,c) - H_2(a,b,c))^2 \quad (2)$$

■ Distancia Manhattan:

$$\sum_{a=0}^{N-1} \sum_{b=0}^{N-1} \sum_{c=0}^{N-1} |H_1(a,b,c) - H_2(a,b,c)| \quad (3)$$

Donde a, b y c representan los tres canales de color (R, G, B o H, S, V), N es el número de píxeles en la imagen y H₁, H₂ son los histogramas de color.

■ Distancia Bhattacharyya aplicada a Histogramas

$$\sum_i \sqrt{R_i} \sqrt{S_i} \quad (4)$$

² Un espacio de color es la especificación de un sistema de coordenadas tridimensional y de un sub-espacio de éste sistema en el que cada color queda representado por un único punto.

Donde R_i y S_i son los histogramas de color e i es el número de píxeles en la imagen.

2.6. Algoritmos de Clasificación

Un algoritmo de clasificación es un es un procedimiento de agrupación en base a algún criterio, generalmente distancia o similitud.

Los algoritmos de clasificación se pueden clasificar en dos grupos: supervisados y no supervisados. Se dice que un algoritmo es supervisado cuando se dispone de una muestra de patrones previamente clasificados en cada una de las categorías a considerar. Dicha muestra se conoce como muestra de entrenamiento, supervisión o de aprendizaje. Por su parte un algoritmo se considera no supervisado cuando no existe dicho conocimiento previo. En este caso se inicia con un universo de patrones sin estructura sobre el cual se debe realizar una clasificación pero el número y la naturaleza de las clases a construir forman parte de las definiciones iniciales para resolver el problema [21].

2.6.1. Algoritmos de clasificación supervisados

La metodología tradicional para la evaluación de problemas supervisados consta fundamentalmente de dos pasos: la aplicación del algoritmo a diversos problemas cuya solución se conoce de antemano y la evaluación de los resultados.

Estos problemas básicamente consisten en que, para clasificar automáticamente una nueva muestra, se tiene en cuenta la información que se pueda extraer de un conjunto de objetos disponibles ya clasificados de los que se selecciona una regla de clasificación o clasificador.

Dentro de estos métodos una de las aproximaciones más conocidas se basa en criterios de vecindad, bajo esta perspectiva, los métodos de clasificación exigen la definición de una medida de similitud (o distancia) entre los distintos elementos del espacio de representación. La ventaja de los métodos basados en criterios de vecindad con respecto a otros métodos de clasificación hace referencia a su simplicidad conceptual, que se puede resumir del siguiente modo: La clasificación de un nuevo punto del espacio de representación se calcula en función de las clases, conocidas de antemano, de los puntos más próximos a él [21].

2.6.2. Ejemplos de algoritmos de clasificación supervisados

2.6.2.1. Algoritmo C4.5

Este construye un árbol de decisión desde un grupo de entrenamiento utilizando la estrategia de "profundidad primero", el cual contiene en cada uno de sus vectores contiene las distintas categorías de las muestras y la clase a la que pertenece dicha muestra.

En cada nodo del árbol, el algoritmo prueba todas las características que dividan el conjunto de datos y selecciona el que resulte en la mayor ganancia de información. Su criterio es el normalizado para ganancia de información (diferencia de entropía) que resulta de la elección de un atributo para dividir los datos. El atributo con mayor ganancia de información normalizada es elegido como parámetro de decisión, y luego se vuelve a aplicar recursivamente en subclases cada vez más pequeñas [22].

Su pseudocódigo es el siguiente:

```
Comprobar los casos base
Para cada atributo a
    Encontrar la ganancia de información normalizada de la división de a
    Dejar que  $a\_best$  sea el atributo con la ganancia de información normalizada más alta
    Crear un nodo de decisión que divida  $a\_best$ 
    Repetir en las subclases obtenidas por división de  $a\_best$ , y agregar estos nodos como hijos de nodo
```

2.6.2.2. Algoritmo ID-3 (Induction Decision Trees)

Desarrollado por J. Ross Quinlan en 1983, pertenece a la familia "TDIDT" (*Top-down induction of decision trees*). Crea un árbol de decisión como un método para aproximar una función objetivo de valores discretos que es resistente al ruido de los datos. El resultado puede expresarse como un conjunto de reglas: Si-entonces.

En cada momento y mediante una heurística determinada las variables que portan información relevante para la solución del problema.

Sus inconvenientes son: que favorece a los atributos que tienen muchos valores, los cuales pueden no ser los más útiles, maneja los valores de las variables de forma discreta, genera grandes árboles de decisión que no garantizan reglas eficientes [22].

Su pseudocódigo es el siguiente:

```
Id3(Ejemplos, Atributo-objetivo, Atributos)
    Si todos los ejemplos son positivos devolver un nodo positivo
    Si todos los ejemplos son negativos devolver un nodo negativo
    Si Atributos está vacío devolver el voto mayoritario del valor del atributo objetivo en ejemplos
    En otro caso
        Sea A Atributo el MEJOR de atributos
```

Para cada v valor del atributo hacer

Sea $Ejemplos(v)$ el subconjunto de ejemplos cuyo valor de atributo A es v

Si $Ejemplos(v)$ está vacío devolver un nodo con el voto mayoritario del atributo objetivo de $Ejemplos$

Sino Devolver $Id3(Ejemplos(v), Atributo-objetivo, Atributos/\{A\})$

2.6.2.3. Redes neuronales

Son un método de aprendizaje supervisado inspirado en el funcionamiento del sistema nervioso de los animales. Es una red de unidades llamadas "neuronas" interconectadas que trabajan juntas para producir un estímulo de salida. Las primeras redes neuronales datan de 1943 por los neurólogos McCulloch y Pitts. [23][24]

Cada neurona recibe una entrada a través de interconexiones y emite una salida dada por tres funciones:

Una función de propagación o de excitación que por lo general es la sumatoria de todas las entradas, por el peso de su conexión, si el resultado es positivo se dice que la conexión es excitatoria y si es negativo se dice que es inhibidora.

Una función de activación que modifica la anterior (puede no existir).

Una función de transferencia que se aplica al valor devuelto por la función anterior, se usa para interpretar el resultado, algunas de las más usadas son la función sigmoidea (para que el resultado este entre $[0,1]$) o la tangente hiperbólica (entre $[-1$ y $1]$).

2.6.3. Algoritmos de clasificación no supervisados

A diferencia de los problemas supervisados, para los problemas no supervisados no existe una forma explicativa de evaluar sus resultados, por lo cual es necesario hacer un "análisis de grupos" que se define como la agrupación de patrones dentro de grupos homogéneos según una medida de similitud. [21]

2.6.3.1. Algoritmos de agrupamiento (*clustering*)

Clasifican los datos en base a algún criterio el cual suele ser la distancia. Se basa en la teoría de que muestras de un mismo grupo tendrán características en común. Y el conjunto de dichos grupos forma el total del espacio de datos. [25]

A pesar de que su objetivo es el mismo, los algoritmos de agrupamiento pueden ser separados en base a su método de agrupación: por conexión, por centroides, por distribución o por densidad

2.6.3.1.1. Modelos de conexión

Se basa en la distancia entre las conexiones. Este se basa en la teoría de que, los objetos más cercanos están más relacionados entre sí, que con los objetos más lejanos. Para establecer la distancia que se usará para determinar si los datos están o no relacionados se pueden usar varios métodos por ejemplo el mínimo de las distancias entre todos los objetos, el máximo de las distancias entre los objetos, o el promedio de dichas distancias. Estos algoritmos pueden iniciar con un solo grupo que posteriormente es dividido (divisivo), o con cada dato en un grupo individual los cuales se van fusionando (aglomerado). A pesar de ser fáciles de entender, suelen ser difíciles de interpretar, y tienen un grado de complejidad muy alto lo cual hace que sean lentos para grupos de datos muy grandes.

2.6.3.1.2. Modelos de centroides

En estos modelos, los clúster se ven representados por un vector central, el cual no debe ser necesariamente un miembro de los grupos de datos. Cuando se tiene definido el número de grupos a encontrar (k) el problema se puede definir formalmente como: encontrar los centros para cada grupo (k) y asignarle los objetos más cercanos.

Estos modelos tienden a ser problemas NP completos (un problema que, dada su complejidad no se puede resolver de forma óptima en un tiempo razonable) por lo que su solución es solo una aproximación. El modelo de clasificación más común es el algoritmo "k-means"

La mayoría de estos algoritmos requieren que se sepa previamente la cantidad de grupos que hay en el espacio de pruebas, lo cual es considerado uno de sus mayores problemas, además de que tiende a tener errores de clasificación en los bordes de las clases. [26] [27]

2.6.3.1.3. Modelos de distribución

Están muy relacionados con la estadística, los grupos son definidos como objetos que tienen mayor posibilidad de pertenecer a un mismo grupo. A pesar de que tienden a ser buenos, tienen como problema que si se incrementa la complejidad con la que se clasifica, siempre se mejoraran los resultados, lo cual hace que sea difícil escoger un nivel de complejidad adecuado.

Los modelos de distribución son semánticamente fuertes, ya que no solo nos dan una clasificación sino que además se puede observar la correlación y dependencia entre distintos atributos. Su mayor dificultad es escoger un modelo de datos apropiado para optimizarlo, ya que para muchos modelos de datos reales no existe un algoritmo matemático que modele los datos de forma correcta.

Uno de los modelos de distribución más comunes es la mezcla de modelos gaussianos utilizando el algoritmo de esperanza-maximización.

2.6.3.1.4. Modelos de densidad

En estos los grupos son definidos según la densidad en el espacio de pruebas, y los objetos en las partes donde se dividen los grupos tienden a considerarse "ruido". Uno de sus problemas es que dependen de que la densidad en algún punto del espacio de pruebas se reduzca para que sean capaces de identificar 2 grupos, por lo que tienden a fallar con "datos de la vida real" [26].

2.6.4. Ejemplos de Algoritmos de clasificación no supervisados

2.6.4.1. K-means

Este algoritmo tiene como fundamento la partición de un conjunto de n muestras en k grupos en el que cada observación pertenece al grupo más cercano a la media. Este problema es computacionalmente difícil (NP-complejo)

Suponga que tenemos n características en vectores $x_1, x_2, \dots, x_n$, donde cada x_i está representado en un espacio m dimensional y sabemos que están agrupados en k grupos ($k < n$). Definimos a m_j como la media del j -ésimo grupo. Si los grupos están bien separados, podemos usar una mínima distancia de clasificación para separarlos. Esto es, podemos decir que x_i está en el j -ésimo grupo si $\|x_i - m_j\|$ es el mínimo con respecto a los k grupos. Su pseudocódigo es el siguiente:

```
Haga una estimación inicial para la  $k$  medias  $m_1, m_2, \dots, m_k$ .
Mientras no cambie alguna media:
    Use la media estimada para clasificar los datos en grupos.  $b(i,j) = 1$  si el  $i$ -ésimo
    dato, es el más cercano a la  $j$ -ésima media.
    Para cada uno de los grupos
        Calcule la nueva media  $m_i$ , utilizando la nueva clasificación
    Fin
Fin
```

2.6.4.2. Algoritmo de esperanza-maximización (Algoritmo EM)

Inicia "adivinando" los parámetros de la distribución de nuestros objetos; dichos parámetros son usados para calcular las probabilidades de que un objeto pertenezca a un grupo, y en base a esas probabilidades recalcula los parámetros iniciales hasta que coincidan, aunque a causa de esto tiende a encontrar mínimos locales por lo que es recomendable correr este algoritmo varias veces [27]

2.6.4.3. DBSCAN

Este algoritmo puede identificar grupos en muestras de datos analizando la densidad de los datos en el espacio de prueba utilizando muy pocos parámetros, es decir se pueden agrupar los datos incluso si se tiene muy poco conocimiento previo de dichos datos, además de que es capaz de identificar "ruido" dentro de nuestros datos o los datos que delimitan 2 grupos distintos. [26]

Este algoritmo se rige por las siguientes reglas:

1. Vecindad Eps: Para que un punto pertenezca a un grupo, debe haber por lo menos otro punto con una distancia menor a una distancia determinada (se le llama Eps).
2. En alcance de densidad directa: Hay 2 tipos de puntos, los puntos borde, y los puntos del núcleo, un punto del núcleo tiene más puntos en vecindad Eps, que un punto borde.
3. En alcance de densidad: un punto pertenece a cierto grupo, si hay un punto que cumple con la regla 1 y 2 y hay una cadena de puntos que cumplan la regla 2 que los una.
4. Conectados por densidad: en ocasiones 2 puntos de borde no estarán en alcance de densidad, pero ambos pertenecen al mismo grupo si hay un punto de núcleo con el cual los dos puntos borde están en alcance de densidad.
5. Si el punto p pertenece al grupo C y q es un punto en alcance de densidad del punto p y conectados por un número máximo de puntos, entonces q pertenece también al grupo C
6. Los puntos que no pertenezcan a ningún grupo, son ruido

2.7. Metodología

Consideramos las Metodologías Ágiles, ya que estas permiten reducir drásticamente los tiempos de desarrollo sin renunciar a las prácticas esenciales de la Ingeniería de Software, asegurando de esta forma la calidad del producto. De las metodologías ágiles trabajaremos específicamente con Programación Extrema.

La Programación Extrema (XP) es la primera metodología ágil y posiblemente el método ágil más conocido y ampliamente utilizado. El nombre fue acuñado por Beck, debido a que el enfoque fue desarrollado utilizando buenas prácticas reconocidas, como el desarrollo iterativo, y con la participación del cliente en niveles "extremos" [10]. Está centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software,

promoviendo el trabajo en equipo y preocupándose por el aprendizaje de los desarrolladores y propiciado un buen clima de trabajo [11]. XP se basa en:

- La retroalimentación continua entre el cliente y el equipo de desarrollo.
- La comunicación fluida entre todos los participantes.
- La simplicidad en las soluciones implementadas.
- El coraje para enfrentar los cambios.

La programación extrema es especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes en los que existe un alto riesgo técnico.

La programación extrema abarca un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades del marco de trabajo: planeación, diseño, codificación y pruebas. En la figura 6 (tomada de [12]) se ilustra el proceso de la programación extrema y se observan algunas de las ideas y tareas clave asociadas con cada actividad del marco de trabajo [13].

El proceso en la programación extrema consiste (a grandes rasgos) en [11]:

- El cliente define una acción, mediante la redacción de una historia de usuario, que debe realizar el sistema y le asigna un valor de implementación.
- Se crean tarjetas CRC (Colaborador-Responsable-Clase) para identificar y organizar cada una de las clases orientadas a objetos que componen esta acción.
- La pareja de desarrolladores estima el esfuerzo necesario para la codificación e implementa la acción.
- Se realizan pruebas de unidad o de aceptación según sea el caso.
- Vuelve al paso uno.

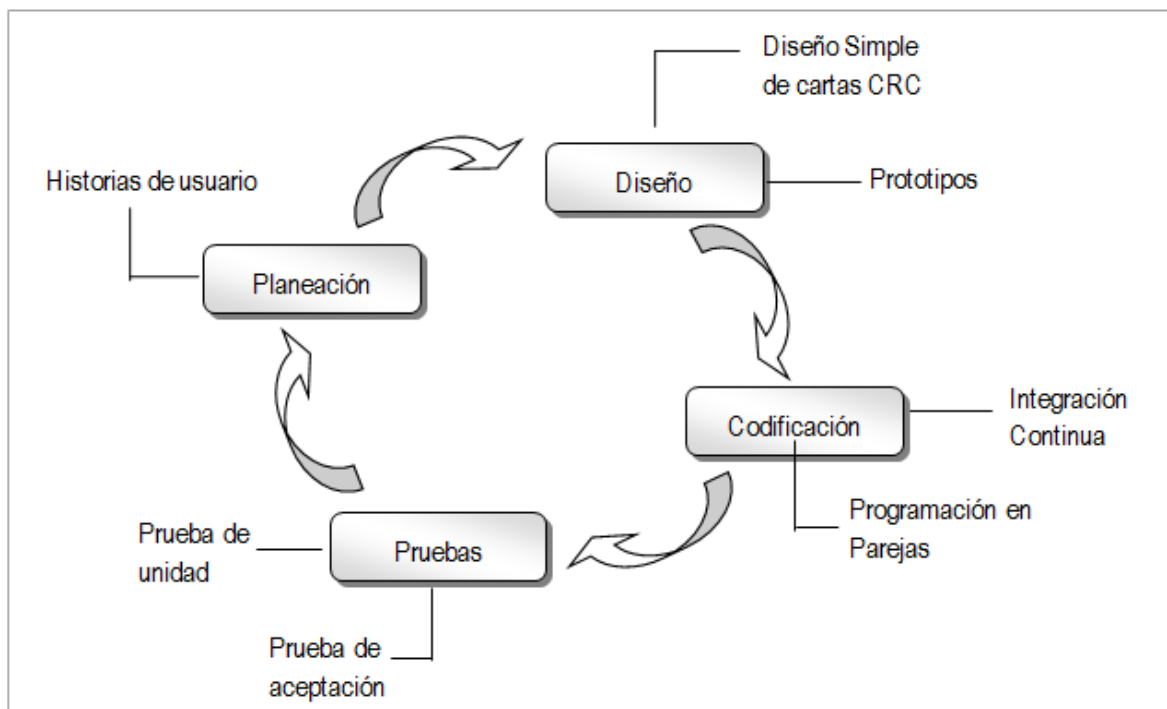


Figura 6. Proceso de la Programación Extrema.

3. Propuesta de solución

Desarrollar un sistema de clasificación de imágenes por estilo artístico. Para esto se analizarán los descriptores de bajo nivel a fin de construir un vector característico con el cual a través de algoritmos de multi-clasificación se generará una taxonomía por estilo. Consideramos, como ya se mencionó, principalmente en esta etapa del proyecto, tres estilos de arte: Barroco, Cubismo e Impresionismo.

3.1. Objetivos Específicos

1. El núcleo del trabajo realizado para alcanzar el objetivo propuesto, será entonces el análisis de las características relevantes que consideran los especialistas en historia del arte, esto nos permitirá desarrollar un conjunto de algoritmos para extracción de estas características mediante el análisis de las imágenes pertenecientes a los estilos pictóricos ya mencionados. Cuando nos referimos, a algoritmos de extracción de características, queremos decir que estos algoritmos obtienen las características de bajo nivel de la imagen las cuales pueden interpretarse como características pictóricas cuantificables. Por ejemplo, una característica relevante en las pinturas impresionistas es el manejo de la luz y la carencia de colores oscuros, este hecho se puede cuantificar estudiando el histograma de grises y obteniendo el porcentaje de píxeles oscuros de la imagen. Otro ejemplo es el algoritmo de recuento de borde el cual podría informar cuántos bordes definidos están presentes en la imagen (línea). Estos datos cuantitativos de la pintura se organizan y con ellos se alimenta un algoritmo de aprendizaje
2. Una vez extraídas las características se requiere de la utilización de algoritmos de aprendizaje supervisado para llevar a cabo la categorización de estilos. Los algoritmos de clasificación, utilizan esta información para aprender a discriminar entre los estilos de arte, por ejemplo, podría notar que las pinturas con una amplia variedad de colores tienden a ser expresionista abstracta. En este proyecto estudiaremos particularmente los algoritmos de clasificación k-means y árboles de decisión

Este trabajo se centrará en la extracción de las características relacionadas con las características siguientes que permiten cuantificar las características fundamentales en cuanto a iluminación, oscuridad y trazo de los diferentes estilos pictóricos considerados [7]:

1.- Porcentaje de píxeles oscuros, los píxeles oscuros están en un rango de 0 a 64 niveles de gris. La relación del número de píxeles oscuros con el tamaño total de la imagen nos da esta característica. Esta característica distingue por ejemplo a las imágenes del realismo pictórico ya que la mayoría de estas imágenes presentan fondo oscuro y las distingue de las piezas impresionistas donde el porcentaje de píxeles oscuro es reducido.

$$\text{Porcentaje de pixeles oscuros} = \frac{\text{Total de número de pixeles oscuros}}{\text{Total de la imagen}}$$

2. Desviación de la intensidad media local vs la intensidad media global: La imagen se divide en 9 bloques, y la media de las intensidades de cada uno de los bloques se calcula. La desviación se obtiene por la sumatoria de las n diferencias, de cada uno de los medios locales con respecto a la de la media global, donde n es el número de bloques

$$\text{Desviacion} = \sum_{i=1}^9 (\text{promedio}_i - \text{promedio})$$

En la ecuación promedio_i representa el valor de luminancia del i -ésimo bloque, promedio es el valor de luminancia promedio de la imagen completa y calculado por la siguiente ecuación:

$$\text{promedio} = \frac{\sum_{i=1}^r \sum_{j=1}^c f_{ij}}{rc}$$

3. Valor máximo en el histograma de luminancia: Este fue encontrado por el cálculo de los máximos del histograma de luminancia en 256 bins. Esta característica ayuda de nuevo en separar al impresionismo del realismo. Las imágenes impresionistas son generalmente más brillantes con un máximo en los rangos superiores de la escala de grises, a diferencia de realismo donde las imágenes son generalmente oscuras con un máximo en los intervalos inferiores.

$$\text{Rango de colores} = \max_i |H(i)|, \quad i = 0, 1, \dots, 255$$

Donde $H(i)$ es el valor del histograma para i -ésimo nivel de gris.

4. Número de máximos locales en el histograma de luminancia: Esta función es muy útil para distinguir pinturas impresionistas donde por lo general hay un o varios máximos

5. Coeficiente de gradiente calculado a partir del mapa de degradado de la imagen de la pintura.

$$\text{Coeficiente de gradiente} = \text{Normalizado} \left(\sum_{i=1}^r \sum_{j=1}^c \sqrt{(f_{ij})_x^2 + (f_{ij})_y^2} \right)$$

Donde r y c son los números de las filas y columnas de la imagen respectivamente, $(f_{ij})_x$ y $(f_{ij})_y$ son las derivadas de primer orden de los pixeles (i, j) de la imagen en las direcciones x y y respectivamente

6. "Skew", es la desviación de la distribución de nivel de gris de la distribución de Gauss. Generalmente las pinturas con las características del impresionismo tienen Skew negativo.

$$\text{Skew} = \frac{\sum_{i=1}^r \sum_{j=1}^c \left(\frac{f_{ij} - \text{promedio}}{\sigma} \right)^3}{rc}$$

donde σ está definida de la siguiente manera:

$$\sigma = \frac{\sqrt{\sum_{i=1}^r \sum_{j=1}^c (f_{ij} - \text{promedio})^2}}{rc - 1}$$

Dentro de la clasificación de obras de arte por estilo artístico nos encontramos con 2 tendencias. Una de estas, desarrollada por Thomas Edward Lombardi [8], se enfoca en el análisis de los colores, tipos de líneas, la iluminación y las texturas que se observan en la imagen, enfocándose principalmente en los autocorrelogramas de color y la distribución espacial de los histogramas cromáticos para su clasificación.

La segunda tendencia, desarrollada por Oguz Icoglu [7] utiliza un vector de 6 dimensiones que presentan las características más importantes para la clasificación de un estilo artístico. Este método convierte inicialmente la imagen a escala de grises inicialmente para poder obtener el porcentaje de colores oscuros en la imagen, su coeficiente de gradiente, el máximo local y global de iluminación dentro del histograma (de grises), la derivación del gris promedio de cada una de n partes iguales de la imagen en base al promedio global del histograma de grises y el "Skew" que es la derivación de la distribución de grises de su distribución de Gauss

Una vez extraídas las características de las imágenes, se requiere el uso de métodos de clasificación que procesen estas características, los métodos cuya eficiencia se desea comparar son algoritmo K-means, el método adaptativo y el basado en árboles de decisión.

El algoritmo K-means es un método de agrupamiento heurístico con un número de clases conocido (K), basado en la minimización de las distancias de cada una de las muestras, con el centro del grupo al que fue asignado. Es un algoritmo sencillo y eficiente además de que al ser secuencial requiere un mínimo de memoria. Sin embargo se ve afectado por el orden en el que se presenten los patrones, donde los primeros en ser presentados determinaran la configuración final y su comportamiento depende enormemente del parámetro (K).

El algoritmo C4.5 construye un árbol de decisión desde un grupo de entrenamiento, el cual para cada uno de los nodos del árbol, el algoritmo escoge la característica que divida de forma más eficiente el conjunto de muestra en subconjuntos. Su criterio es el normalizado para ganancia de información (diferencia de entropía) que resulta de la elección de un atributo para dividir los datos. El atributo con mayor ganancia de información normalizada es elegido como parámetro de decisión, y luego se vuelve a aplicar recursivamente en subclases cada vez más pequeñas.

3.2. Construcción de la base de datos de imágenes de obras pictóricas

Las imágenes que constituyen la base de datos de imágenes de prueba se construyó utilizando la Enciclopedia de Artes Visuales Wikipaintings

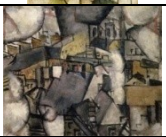
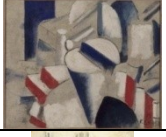
<http://www.wikipaintings.org/en/paintings-by-style>.

Esta enciclopedia fue construida por expertos en el área por lo que la BD considerada ya está validada. La base de datos de prueba que se uso consiste de 450 imágenes, 150 barrocas, 150 cubistas y 150 impresionistas de algunos de los pintores más representativos de su estilo.

Para conseguir los objetivos considerados en la sección precedente, se planean las actividades siguientes:

1. Desarrollar las funciones mediante las cuales se extraerán las características de bajo nivel de las imágenes a clasificar.
2. Validar los resultados de nuestras funciones contra los resultados del documento base.
3. Investigación sobre el funcionamiento de R, Statistica y Weka.
4. Análisis y comparación de los resultados obtenidos mediante las herramientas de clasificación.

Inicialmente se recopilaron obras de arte de distintos estilos artísticos previamente clasificados por un experto así como varias características de éstos a fin de crear una base de datos con dichas obras. Un subconjunto de las pinturas consideradas se muestra en la tabla 1 siguiente.

Autor	Titulo	Estilo	Imagen
Leger	The Woman in Blue	Cubismo	
Leger	The level Crossing	Cubismo	
Leger	Chimneys on Rooftops	Cubismo	
Leger	Trouville	Cubismo	
Leger	Contrasts of Forms	Cubismo	
Gris	Tres lámparas	Cubismo	

Gris	El Sifón	Cubismo	
Gris	Fantomas	Cubismo	
Gris	La Ventana Abierta	Cubismo	
Gris	Retrato de Picasso	Cubismo	
Picasso	Composition "Peasants"	Cubismo	
Picasso	A girl from Avignon	Cubismo	
Picasso	Bust of young woman from Avignon	Cubismo	
Picasso	Composition with skull (study)	Cubismo	
Picasso	Female bust	Cubismo	
Caravaggio	Saint Francis of Assisi in Ecstasy	Barroco	
Caravaggio	David and Goliath	Barroco	
Caravaggio	Saint Catherine of Alexandria	Barroco	
Caravaggio	Sacrifice of Isaac	Barroco	
Caravaggio	Martha and Mary Magdalene	Barroco	

Reni	Assumption of the Virgin	Barroco	
Reni	Assumption	Barroco	
Reni	Rosary Madonna	Barroco	
Reni	St. Peter Penitent	Barroco	
Reni	David with the Head of Goliath	Barroco	
Velazquez	Don Cristobal Suarez de Ribera	Barroco	
Velazquez	Head of a Man	Barroco	
Velazquez	An Old Woman Cooking Eggs	Barroco	
Velazquez	Breakfast	Barroco	
Velazquez	Kitchen Scene with the Supper in Emmaus	Barroco	
Pissarro	The Lock at Pontoise	Impresionismo	
Pissarro	Village Corner	Impresionismo	
Pissarro	Avenue de l'Opera, Place du Theatre Francais. Misty	Impresionismo	
Pissarro	The Port of Le Havre	Impresionismo	

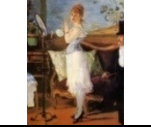
Pissarro	Place du Theatre Francais	Impresionismo	
Degas	Ballet Dancers in the Wings	Impresionismo	
Degas	In the Dance Studio	Impresionismo	
Degas	Dancers tying shoes	Impresionismo	
Degas	Two Ballet Dancers	Impresionismo	
Degas	Place de la Concorde	Impresionismo	
Manet	La familia de Monet	Impresionismo	
Manet	Claude Monet pintando en su barco estudio	Impresionismo	
Manet	El Ferrocarril	Impresionismo	
Manet	El Gran Canal de Venecia	Impresionismo	
Manet	Nana	Impresionismo	

Tabla 1

Las características de bajo nivel que se usaron son las siguientes:

Intensidad o iluminación, que está definida por la siguiente ecuación:

$$L = 0.299R + 0.587G + 0.114B$$

Y de esta obtenemos 5 sub-características:

1. Porcentaje de colores oscuros, es decir con una iluminación entre 1 y 64
2. Desviación de la intensidad media, contra la intensidad global: las imágenes se dividieron en bloques de los cuales se obtiene la intensidad

local, y de la sumatoria de sus diferencias contra la desviación global fue que se obtuvo esta característica

3. El valor máximo en el histograma de iluminación.
4. El número de máximos locales en el histograma de iluminación
5. El coeficiente del mapa de gradientes. El gradiente horizontal y vertical de cada píxel es calculado y luego el vector resultante es sumado

$$g = \sum_r \sum_c \sqrt{f(i,j)_x^2 + f(i,j)_y^2}$$

Detector de bordes mediante 2 características:

1. La función nativa de matlab `bwconncomp(-)` la cual cuenta los bordes en una imagen en escala de grises
2. El radio del número de bordes en la escala de grises, entre el número de bordes en dicha imagen en escala de rojos, verdes y azules

$$radio = e / (r + g + b)$$

Color en (R,G,B) del que se obtienen las siguientes 3 características:

1. Alcance de la paleta de color: es el porcentaje de colores únicos dentro de una obra
2. Histograma de Hue
3. Los valores únicos RGBXY: si una imagen esta formada por $M \times N \times 3$ canales, se crea una matriz $M \times N$ y 5 columnas, para los valores R, G, B, X y Y de cada uno de los píxeles, se calcula su matriz covarianza obteniendo una matriz de 5×5 cuyos valores individuales describen la variación en el espacio de colores RGB y su variación en el espacio.

Y la textura, usando las características de Haralick que corresponden a las pinceladas en la obra:

1. Momento angular

$$f_1 = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} \{p(i,j)\}^2$$

2. Contraste

$$f_2 = \sum_{n=0}^{Ng-1} n^2 \left(\sum_{i=1}^{Ng} \sum_{j=1}^{Ng} (p(i,j)) \right)$$

Donde $|i-j|=n$

3. Correlación

$$f_3 = \frac{\sum_{i=1}^{Ng} \sum_{j=1}^{Ng} (ij) p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}$$

4. Varianza de cuadrados

$$f_4 = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} (i - \mu)^2 p(i, j)$$

5. Diferencia inversa del momento

$$f_5 = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} \frac{1}{1 + (i - j)^2} p(i, j)$$

6. Promedio

$$f_6 = \sum_{i=2}^{2Ng} i p_{x+y}(i)$$

7. Suma de varianza

$$f_7 = \sum_{i=2}^{2Ng} (i - f_8)^2 p_{x+y}(i)$$

8. Suma de entropía

$$f_8 = \sum_{i=2}^{2Ng} p_{x+y}(i) \log\{p_{x+y}(i)\}$$

9. Entropía

$$f_9 = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} p(i, j) \log(p(i, j))$$

10. Diferencia de la varianza

$$f_{10} = E[p_{x-y}^2] - E[p_{x-y}]^2$$

11. Diferencia de entropía

$$f_{11} = \sum_{i=0}^{Ng-1} p_{x-y}(i) \log\{p_{x-y}(i)\}$$

12. Medidas de información de correlación

$$f_{12} = \frac{HXY - HXY1}{\max\{HX, HY\}}$$

$$f_{13} = (1 - \exp[-2.0(HXY2 - HXY)])^{1/2}$$

13. Coeficiente máximo de correlación

$$f_{14} = (\text{Segundo mayor valor propio de } Q)^{1/2}$$

Donde:

$$Q(i, j) = \sum_{k=1}^{Ng} \frac{p(i, k)p(j, k)}{p_x(i)p_y(k)}$$

Para la implementación de dichas características se eligió Matlab dado que en este lenguaje se puede trabajar muy fácilmente con imágenes que es la parte fundamental del trabajo. El código implementado para cada característica es el siguiente:

3.3. Características de iluminación

```
function [L,Iindex,binvalor,mediaig,MBL,media,HL,contraste,bcontraste]
= CaracIntencidad5 (ImRGB)
[imR,imG,imB]=DescomponerRGB (ImRGB);
Lu=(0.299 * imR) + (0.599*imG)+(0.114*imB);
[L]= estandariza (Lu,9);
L=floor (L);
L=double (L);
oscuro=pixoscuros (L);
Iindex(1)=oscuro*100;
[gradIX,gradIY]=gradient (L);
[m,n]=size (gradIX);
gradiente=0;
for i=1:m
    for j=1:n
        gradiente = gradiente + sqrt ((gradIX(i,j)^2)+(gradIY(i,j)^2));
    end
end;
gradiente= gradiente/(m*n);
Iindex(2)=gradiente;
[HL, bins]= histogramacuantizadohue (L,255);
minaltura=0;
porcentaje=0.50;
minaltura= max (HL) - (max (HL) *porcentaje);
HL2(1)=0;
HL2(2:257)=HL;
HL2(258)=0;
[peaks,locs]=findpeaks (HL2,'minpeakdistance',93,'MINPEAKHEIGHT',minaltura);
numpicos=size (peaks,2);
maximo=0;
for i=1:numpicos
    binvalor(i,1)= locs(1,i)-2 ;
    binvalor(i,2)=peaks(1,i);
    if maximo < binvalor(i,2)
        maximo = binvalor(i,2);
        colorvalue = binvalor(i,1);
    end;
end;
Iindex(3)=numpicos;
Iindex(4)=colorvalue;
mediaig=mean (mean (L));
[MBL, media] = Block (L,3);
[m,n]=size (MBL);
k=0;
suma=0;
for i=1:m
    for j=1:n
        k= k +1;
        contraste(k)= abs ((MBL(i,j)-mediaig));
        suma= suma+ contraste(k);
    end
end;
end;
```

```

for k=1:3
    bcontraste(k)=0;
end;
Iindex(5)=(suma/((m*n)-1));
for k=1:9
    if (contraste(k) < Iindex(5)), bcontraste(1)= bcontraste(1)+1;
    elseif (contraste(k) >= Iindex(5) && contraste(k)<
(2*Iindex(5))), bcontraste(2)= bcontraste(2)+1;
    elseif (contraste(k) >= 2*Iindex(5) && contraste(k)<
(3*Iindex(5))), bcontraste(3)= bcontraste(3)+1;
    end;
Iindex(6)=skewness(HL);
end

```

3.4. Características de textura

```

function [ F ] = haralick( glcm )
S=size(glcm,1);
f_2=zeros(S);
f_3=zeros(S);
f_4=zeros(S);
f_5=zeros(S);
f_6=zeros(1,2*S);
f_7=zeros(1,2*S);
f_8=zeros(1,2*S);
f_9=zeros(S);
f_11=zeros(1,S);
pxy=zeros(1,2*S);
px_y=zeros(1,S);
HX_=zeros(1,S);
HY_=zeros(1,S);
HXY_1=zeros(S);
HXY_2=zeros(S);
M = glcm/sum(glcm(:));
f_1 = M.^2;
f1 = sum(f_1(:));
u = mean2(M);
py = sum(M,1);
px = sum(M,2);
for i=1:S
    for j=1:S
        f_3(i,j) = i*j*M(i,j);
        f_4(i,j) = (i-u)^2*M(i,j);
        f_5(i,j) = M(i,j)/(1+(i-j)^2);
        f_9(i,j) = M(i,j)*log(M(i,j)+eps);
        pxy(i+j) = pxy(i+j)+M(i,j);
        px_y(abs(i-j)+1) = px_y(abs(i-j)+1)+M(i,j);
        HX_(i)= px(i)*log(px(i)+eps);
        HY_(j)= py(j)*log(py(j)+eps);
        HXY_1(i,j) = M(i,j)*log(px(i)*py(j)+eps);
        HXY_2(i,j) = px(i)*py(j)*log(px(i)*py(j)+eps);
    end
end
ux = mean(px); sx=std(px);
uy = mean(py); sy=std(py);
f3 =(sum(f_3(:))-(ux*uy))/(sx*sy);
f4 = sum(f_4(:));
f5 = sum(f_5(:));
f9 = -sum(f_9(:));
HX = -sum(HX_);

```

```

HY = -sum(HY_);
HXY = f9;
HXY1 = -sum(HXY_1(:));
HXY2 = -sum(HXY_2(:));
f12 = (HXY-HXY1)/max([HX, HY]);
f13 = (1 - exp((-2)*(HXY2 - HXY)))^0.5;
for i=2:2*S
    f_6(i) = i*pxy(i);
    f_8(i) = pxy(i)*log(pxy(i)+eps);
end
f6 = sum(f_6);
f8 = -sum(f_8);
for i=2:2*S
    f_7(i)=(i-f8)^2*pxy(i);
end
f7 = sum(f_7);
f10 = var(px_y);
for k=1:S
    f_2(k) = (k-1)^2*px_y(k);
    f_11(k) = px_y(k)*log(px_y(k)+eps);
end
f2 = sum(f_2(:));
f11 = -sum(f_11);
F = [f1;f2;f3;f4;f5;f6;f7;f8;f9;f10;f11;f12;f13];

end

```

3.5. Características de los bordes

```

function [totallineas]= bordes (ImRGB)
[imR,imG,imB]=DescomponerRGB(ImRGB);
[LineasE,thresE,E]=cannybordes(ImRGB, .4,3.5,1);
[LineasR,thresR,R]=cannybordes(imR, .4,3.5,1);
[LineasB,thresB,B]=cannybordes(imB, .4,3.5,1);
[LineasG,thresG,G]=cannybordes(imG, .4,3.5,1);
totallineas = E/(R+B+G);
end

```

4. Resultados obtenidos

Se realizó un análisis sobre imágenes de pinturas con el fin de obtener un algoritmo de decisión para distinguir entre diferentes periodos artísticos. Para realizar el análisis se llevaron a cabo técnicas de análisis de imágenes y minería de datos.

Se consideraron inicialmente 80 diferentes valores para cada una de las 450 imágenes de entrenamiento pertenecientes a tres periodos artísticos diferentes: 150 del barroco, 150 del cubismo y 150 del impresionismo. Entre estas características encontramos iluminación, oscuridad, línea, conteo de colores (paleta de 56 colores) y características de la distribución de los colores. Las 80 variables son numéricas de tipo continuo y discreto.

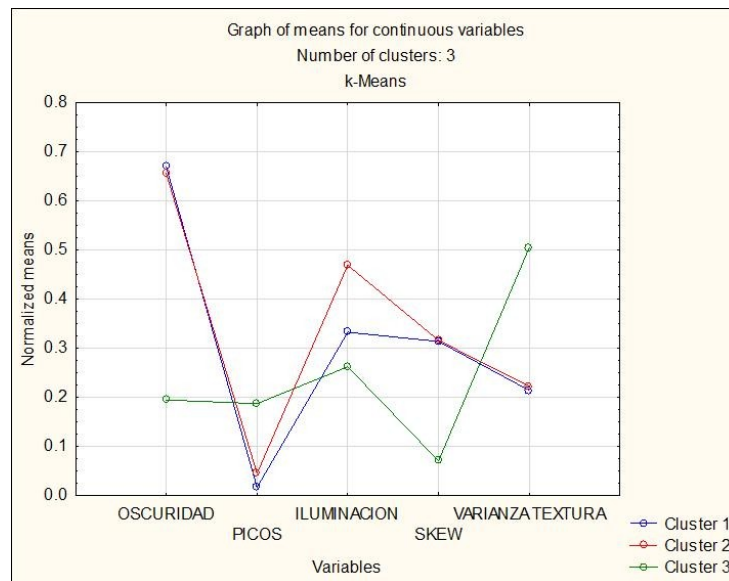
Para poder distinguir entre las variables de interés primero realizamos un análisis exploratorio para eliminar las variables que no presentaran patrones para distinguir entre los diferentes estilos. Las primeras variables que fueron eliminadas fueron aquellas que presentaron variación nula. Las variables que presentaron variación nula fueron los colores (de la variable 17 en adelante), lo cual nos permitió discriminar 40 variables espurias. A la hora de remover variables espurias, escalar la matriz de datos y realizar un agrupamiento, pudimos observar ligeros patrones.

Para distinguir patrones de similitud en las características de las pinturas realizamos un análisis de clúster con el resto de las variables que quedaron, obteniendo una clara distinción entre el barroco de los otros dos estilos, sin embargo, parece ser más complicado distinguir entre cubismo e impresionismo. Observamos que con la influencia de todas las variables podemos distinguir entre estilos, con un poco de dificultad en el cubismo. En el análisis el 41% de la varianza es explicada por los primeros dos componentes y cerca del 25% únicamente por el primero. Las tres variables con mayor aporte a la varianza del primer componente son la Entropía, la suma de entropía y la iluminación media. Esto parece ser un resultado muy natural, ya que en general, las pinturas barrocas presentan menor iluminación que las pinturas cubistas e impresionistas. La interpretación de la entropía de la pintura es complicada así que se prefiere la iluminación media.

Otra de las limitantes encontradas, es que buena parte, de los estilos pictóricos tienen asociados temas que permiten a los estudiosos de la materia poder clasificarlas con precisión, por ejemplo es común encontrar temas religiosos en pinturas barrocas así como retratos de personajes ya sea de la monarquía o de la iglesia. La pintura barroca está dominada por el movimiento, ya sea de carácter físico, emocional o espiritual; posee colores vivos y a menudo dramáticos contrastes entre luz y sombras. Son características definitivas del Barroco el convencer, transformar o engañar mediante ilusionismo. En el estilo barroco se cultivaron diferentes tipos de géneros bodegones, escenas de género, paisajes así como temas alegóricos. Para ejemplificar el hecho de la necesidad de utilizar características visuales e iconográficas, contrastamos en uno de los experimentos obras de estilo Barroco vs Romanticismo vs Cubista Analítico. Para la muestra considerada, de piezas del Romanticismo, incluimos solamente pinturas de Francisco de Goya que por sus características de color asemejan aquellas del barroco; La elección de piezas cubistas analíticas se debe a que la pintura es casi monocroma en gris y ocre. Los colores en este momento no interesaban, en contraste con el romanticismo y el barroco, pues lo importante eran los diferentes puntos de vista y la geometrización, no el cromatismo. Los artistas cubistas analíticos, elaborando un «nuevo lenguaje» que analiza la realidad y la descompone en múltiples elementos geométricos. Los puntos de vista se multiplicaron, abandonando definitivamente la unidad del punto de vista de la perspectiva renacentista y se introducen en la pintura los «pasos», definidos como ligeras interrupciones de la línea del contorno.

Con las características antes expuestas aplicando K-means esperábamos un 100% de precisión en el agrupamiento de estas pinturas, de hecho solamente se obtuvo una pieza que no se clasificó correctamente, en la gráfica 1,

presentamos la gráfica de las medias para los descriptores considerados así como el resumen de resultados.



Gráfica 1. Resultados de la ejecución del algoritmo de clasificación K-Means, utilizando distancia City-block sin validación cruzada. Número de casos 183, error de entrenamiento 0.646188

Centroids for k-means clustering

Cluster	Estilo	Oscuridad	Picos	Iluminación	Skew	Varianza Textura	Número de Casos	Porcentaje
1	Barroco	63.88211	1.033898	23.31845	3.681851	23.32528	118	64.48087
2	Romántico	62.54456	1.090909	30.17430	3.714045	24.02107	33	18.03279
3	Cubismo Analítico	18.73880	1.375000	19.78172	0.270601	46.69948	32	17.48634

Distancia entre clústers:

	Cluster 1	Cluster 2	Cluster 3
Cluster 1	0.000000	1.189950	2.249753
Cluster 2	1.189950	0.000000	2.337239
Cluster 3	2.249753	2.337239	0.000000

Para este experimento se encontró que una pintura barroca estaba clasificada como cubista analítica, en la figura 7, presentamos la pintura de Guido Reni, Drinking Bacchus así como las imágenes de la pintura que se encuentra en el límite del cluster 1(barroco) y la imagen de la pintura más cercana en el cluster 3 (cubismo analítico) a la pintura de Reni.

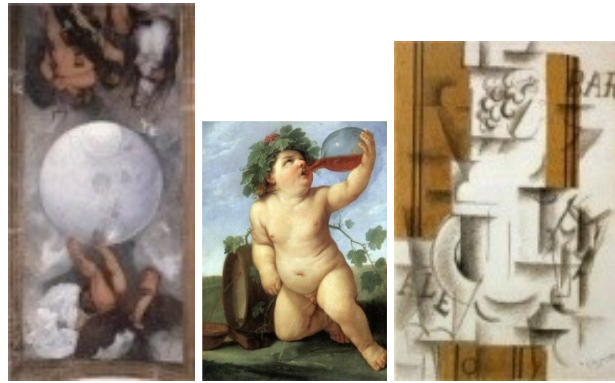
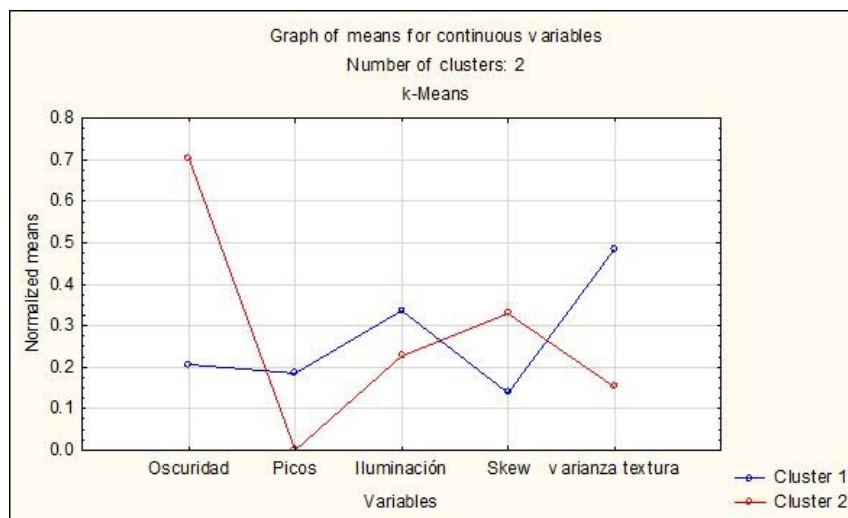


Figura 7. Al centro de la imagen se encuentra la pintura barroca con tema alegórico de Guido Reni, clasificada como Cubista Analítica. Esta franqueada por el lado izquierdo por la pintura de Caravaggio Jupiter, Neptune and Pluto, también de tema alegórico y a la derecha por la pintura de Braque Fruitdish and Glass, de claro estilo cubista analítico. Como puede verse las características de bajo nivel en las tres pueden ser similares pero desde el punto de vista de las características visuales y temáticas un experto nunca clasificaría a la pintura de Reni como Cubista Analítica.

De manera análoga realizamos otro experimento considerando solamente piezas barrocas e impresionistas, en la gráfica 2, podemos comprobar que las variables descriptores utilizadas si permiten hacer una distinción clara entre los dos tipos de estilos. Cabe mencionar que en este experimento el cuadro de Guido Reni que en el experimento anterior fue clasificado como cubista analítico en este experimento es clasificado como impresionista dadas sus características de luminosidad y oscuridad.



Gráfica 2. Resultados de la ejecución del algoritmo de clasificación K-Means, utilizando distancia City-block sin validación cruzada. Número de casos 210, error de entrenamiento 0.666453. En esta gráfica el cluster 1 corresponde a impresionismo y el cluster 2 a barroco, como puede observarse las características de oscuridad son determinantes en el barroco y por otra parte la textura es una característica relevante en el impresionismo no así en el barroco.

Realizamos también un tercer experimento con los tres estilos barroco, impresionista y cubista; sin embargo los resultados no fueron aceptables hecho que atribuimos a la calidad de las imágenes ya que estas no provienen de las piezas originales por los que las características de textura y línea que

permitirían distinguir entre impresionismo y barroco no tienen la calidad requerida.

Para buscar soluciones aceptables considerando los tres estilos pictóricos sobre los que se trabajó en esta investigación decidimos trabajar con el algoritmo C4.5 de árboles de decisión. Este algoritmo considera todas las características "individualmente" para clasificar las muestras, lo que los hace capaces de analizar de forma más flexible cada una de las muestras y unido al uso de muestras de entrenamiento el árbol generado "aprende" que características determinan si una obra pertenece o no a un estilo artístico.

Para determinar cómo se clasificarán las pruebas, el algoritmo C4.5 se basa en el principio de "divide y vencerás", es decir busca las características y divide al conjunto de entrenamiento en los grupos más significativos o con la mayor ganancia de información y después continua de forma recursiva con la estrategia de "profundidad primero" hasta clasificar finalmente las obras.

4.1. Resultados obtenidos utilizando el algoritmo C4.5

Al introducir los datos en Weka y ejecutar el comando para clasificar los datos se obtienen los siguientes resultados:

=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: 150 de todos-weka.filters.unsupervised.attribute.Remove-R1

Instances: 450

Attributes: 25

I1
I2
I3
I4
I5
I6
T1
T2
T3
T4
T5
T6
T7
T8
T9
T10
T11
T12
T13
B1
M1

C1
C2
C3
CLASS

Test mode: evaluate on training data

=== Classifier model (full training set) ===

J48 pruned tree

```
-----
T3 <= 410.154528: BARROCO (90.0/3.0)
T3 > 410.154528
| I6 <= 2.935179
| | I5 <= 14.073962
| | | B1 <= 0.325857: CUBISMO (26.0/1.0)
| | | B1 > 0.325857
| | | | T1 <= 0.153191: CUBISMO (22.0)
| | | | T1 > 0.153191
| | | | | I4 <= 76: CUBISMO (3.0)
| | | | | I4 > 76: IMPRESIONISMO (13.0/3.0)
| | | I5 > 14.073962
| | | | I1 <= 14.09323
| | | | | T3 <= 3661.717417: IMPRESIONISMO (79.0/14.0)
| | | | | T3 > 3661.717417: CUBISMO (3.0)
| | | | I1 > 14.09323
| | | | | I5 <= 33.354159
| | | | | | I1 <= 39.021528
| | | | | | T3 <= 1191.0475
| | | | | | | I4 <= 60
| | | | | | | I5 <= 17.359355: IMPRESIONISMO (2.0)
| | | | | | | I5 > 17.359355: BARROCO (3.0)
| | | | | | | I4 > 60
| | | | | | | I3 <= 1
| | | | | | | I2 <= 8.384981
| | | | | | | | I1 <= 35.722957: CUBISMO (4.0)
| | | | | | | | I1 > 35.722957: IMPRESIONISMO (2.0)
| | | | | | | | I2 > 8.384981: IMPRESIONISMO (11.0)
| | | | | | | I3 > 1: IMPRESIONISMO (3.0)
| | | | | T3 > 1191.0475
| | | | | C1 <= 5
| | | | | | I3 <= 1: CUBISMO (12.0/2.0)
| | | | | | I3 > 1
| | | | | | I6 <= 0.729576
| | | | | | | C3 <= 0
| | | | | | | T4 <= 19.334053: CUBISMO (3.0)
| | | | | | | T4 > 19.334053: IMPRESIONISMO (6.0)
| | | | | | | C3 > 0: IMPRESIONISMO (4.0)
| | | | | | I6 > 0.729576: CUBISMO (5.0)
| | | | | C1 > 5
```

```

| | | | | | | | I6 <= 0.413798
| | | | | | | | | I4 <= 92: IMPRESIONISMO (8.0/1.0)
| | | | | | | | | I4 > 92: CUBISMO (10.0/1.0)
| | | | | | | | | I6 > 0.413798: CUBISMO (25.0/5.0)
| | | | | | | | I1 > 39.021528
| | | | | | | | | T12 <= -0.672787: BARROCO (10.0/1.0)
| | | | | | | | | T12 > -0.672787
| | | | | | | | | T7 <= 27.545729
| | | | | | | | | T5 <= 0.899139: IMPRESIONISMO (3.0)
| | | | | | | | | T5 > 0.899139: CUBISMO (6.0/1.0)
| | | | | | | | | T7 > 27.545729
| | | | | | | | | T13 <= 0.933525: CUBISMO (5.0)
| | | | | | | | | T13 > 0.933525
| | | | | | | | | C3 <= 0
| | | | | | | | | T4 <= 13.971879: BARROCO (2.0)
| | | | | | | | | T4 > 13.971879: CUBISMO (3.0)
| | | | | | | | | C3 > 0: BARROCO (2.0)
| | | | | | | | I5 > 33.354159
| | | | | | | | | T9 <= 2.577681
| | | | | | | | | C2 <= 2: IMPRESIONISMO (6.0)
| | | | | | | | | C2 > 2
| | | | | | | | | I3 <= 1
| | | | | | | | | I4 <= 229: BARROCO (7.0/1.0)
| | | | | | | | | I4 > 229: IMPRESIONISMO (2.0)
| | | | | | | | | I3 > 1
| | | | | | | | | C2 <= 4
| | | | | | | | | I6 <= 0.441848: IMPRESIONISMO (7.0)
| | | | | | | | | I6 > 0.441848
| | | | | | | | | I2 <= 8.657376: BARROCO (4.0)
| | | | | | | | | I2 > 8.657376: IMPRESIONISMO (3.0/1.0)
| | | | | | | | | C2 > 4: IMPRESIONISMO (4.0)
| | | | | | | | T9 > 2.577681
| | | | | | | | I6 <= 0.881491: IMPRESIONISMO (3.0/1.0)
| | | | | | | | I6 > 0.881491: CUBISMO (3.0)
| | | | | | | I6 > 2.935179
| | | | | | M1 <= 108
| | | | | | | I4 <= 3: CUBISMO (2.0)
| | | | | | | I4 > 3
| | | | | | | T7 <= 39.494959
| | | | | | | B1 <= 0.332542: BARROCO (26.0)
| | | | | | | B1 > 0.332542: CUBISMO (3.0/1.0)
| | | | | | | T7 > 39.494959
| | | | | | | T6 <= 7.320015: IMPRESIONISMO (3.0)
| | | | | | | T6 > 7.320015: BARROCO (4.0/1.0)
| | | | | | M1 > 108: CUBISMO (8.0/1.0)

```

Number of Leaves : 43

Size of the tree : 85

Time taken to build model: 0.12 seconds

=== Evaluation on training set ===

=== Summary ===

Correctly Classified Instances	412	91.5556 %
Incorrectly Classified Instances	38	8.4444 %
Kappa statistic	0.8733	
Mean absolute error	0.0948	
Root mean squared error	0.2177	
Relative absolute error	21.3354 %	
Root relative squared error	46.1902 %	
Total Number of Instances	450	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.947	0.02	0.959	0.947	0.953	0.989	BARROCO
	0.873	0.04	0.916	0.873	0.894	0.966	CUBISMO
	0.927	0.067	0.874	0.927	0.9	0.97	IMPRESIONISMO
Weighted Avg.	0.916	0.042	0.917	0.916	0.916	0.975	

=== Confusion Matrix ===

```
a  b  c  <-- classified as
142  4   4 | a = BARROCO
  3 131 16 | b = CUBISMO
   3   8 139 | c = IMPRESIONISMO
```

El árbol de decisión que se obtuvo se presenta en la figura 8.

A raíz del árbol obtenido se pudo observar que, de las 450 imágenes de obras de arte que consta la base de datos utilizada, 412 (91.55%) pudieron ser clasificadas de forma correcta usando el método J48 de árboles de decisión mientras que solo 38 (8.44%) fueron clasificadas de forma errónea dando como resultado la siguiente matriz de confusión:

BARROCO	CUBISMO	IMPRESIONISMO	
142	4	4	BARROCO
3	131	16	CUBISMO
3	8	139	IMPRESIONISMO

Para evaluar dichos resultados existen distintas medidas de evaluación como la precisión (ecuación 1) que es la proporción de documentos relevantes recuperados de entre todos los documentos recuperados y el recuerdo (ecuación 2) que es la proporción de los documentos relevantes entre todos los elementos recuperados

$$precisión = \frac{(documentos_relevantes) \cap (documentos_recuperados)}{documentos_recuperados} \quad (1)$$

$$precisión_{barroco} = \frac{(150) \cap (148)}{148}$$

$$precisión_{barroco} = \frac{142}{148}$$

$$precisión_{barroco} = 0.9594$$

$$precisión_{cubismo} = \frac{(150) \cap (143)}{143}$$

$$precisión_{cubismo} = \frac{131}{143}$$

$$precisión_{cubismo} = 0.9161$$

$$precisión_{impresionismo} = \frac{(150) \cap (159)}{159}$$

$$precisión_{impresionismo} = \frac{139}{159}$$

$$precisión_{cubismo} = 0.8742$$

$$recuerdo = \frac{(documentos_relevantes) \cap (documentos_recuperados)}{documentos_relevantes} \quad (2)$$

$$recuerdo_{barroco} = \frac{(150) \cap (148)}{150}$$

$$recuerdo_{barroco} = \frac{142}{150}$$

$$recuerdo_{barroco} = 0.9466$$

$$recuerdo_{cubismo} = \frac{(150) \cap (143)}{150}$$

$$recuerdo_{cubismo} = \frac{131}{150}$$

$$recuerdo_{cubismo} = 0.8733$$

$$recuerdo_{impresionismo} = \frac{(150) \cap (159)}{150}$$

$$recuerdo_{impresionismo} = \frac{139}{150}$$

$$recuerdo_{cubismo} = 0.9266$$

5. Conclusiones y trabajos a futuro

Se realizó un análisis sobre imágenes de pinturas con el fin de obtener un algoritmo de decisión para distinguir entre diferentes periodos artísticos. Para realizar el análisis se llevaron a cabo técnicas de análisis de imágenes y minería de datos.

Se consideraron inicialmente 80 diferentes valores para cada una de las 450 imágenes de entrenamiento pertenecientes a tres periodos artísticos diferentes: 150 del barroco, 150 del cubismo y 150 del impresionismo. Entre estas características encontramos iluminación, oscuridad, línea, conteo de colores

(paleta de 56 colores) y características de la distribución de los colores. Las 80 variables son numéricas de tipo continuo y discreto.

Para poder distinguir entre las variables de interés primero realizamos un análisis exploratorio para eliminar las variables que no presentaran patrones para distinguir entre los diferentes estilos. Las primeras variables que fueron eliminadas fueron aquellas que presentaron variación nula. Las variables que presentaron variación nula fueron los colores (de la variable 17 en adelante), lo cual nos permitió discriminar 40 variables espurias. A la hora de remover variables espurias, escalar la matriz de datos y realizar un agrupamiento, pudimos observar ligeros patrones para los estilos considerados utilizando los algoritmos k-means y árboles de decisión.

Finalmente podemos concluir que se ha llevado un análisis completo y comprensión de datos a partir de técnicas de análisis de imagen y minería de datos. En lo relacionado a la clasificación de imágenes por estilo pictórico podemos concluir que las características consideradas para la distinción entre barroco vs impresionismo por una parte y barroco vs cubismo vs romanticismo son acertadas y se requiere trabajar en la reducción de variables para poder optimizar los resultados. Debido a que se trata de un análisis muy subjetivo, consideramos que los resultados son aceptables.

Como trabajo futuro consideramos que se requiere considerar nuevos algoritmos para determinar la *pincelada* que puede servir como punto de diferencia entre estilos conflictivos como es el cubismo y el impresionismo, ya que las características extraídas como ya lo mencionamos no permiten identificar los estilos; claro está que para este tipo de estudios se deberá también mejorar la calidad de las imágenes analizadas.

6. Bibliografía

1. Darshak G. Thakore¹, A. I. Trivedi, *Content based image retrieval techniques – Issues, analysis and the state of the art.*, University Vadodara Gujarat.
2. Peter L. Stanchev, David Green Jr., Boyan Dimitrov (2006), Some Issues in the Art Image Database Systems, *Journal of Digital Information Management*, Vol 4. Num. 4, 227-232.
3. Gevers Th. Smeulders A.W.M. (2004). *Content-based Image Retrieval: An Overview, Survey on content-based image retrieval*. En: Emerging Topics in Computer Vision, Medioni G. & Kang S. B. (Eds.). Prentice Hall.
4. Smeulders Arnold W. M. , Hardman Lynda, Schreiber Augustus T., Geusebroek Jan-Mark (2002). *An integrated multimedia approach to cultural heritage e-documents*. ACM Workshop on Multimedia Information Retrieval. ACM Press.
5. E. H. Gombrich (1983). *Arte Percepción y Realidad*. España: Paidós.
6. Erwin Panofsky (1999), *La perspectiva como forma simbólica*, España: Tusquets.

7. D. Pablo Gil Vázquez (2008). *Estrategias para identificar oclusiones y planificación monocular para una mejora de la percepción visual de la escena*. Tesis Doctoral, Universidad de Alicante. Alicante, España.
8. P. Gil, F. Torres, F. G. Ortiz (2004). Detección de objetos por segmentación multinivel combinada de espacios de color. *XXV Jornadas de Automática*.
9. Smith A. R. (1978). Color gamut transformation pairs. *SIGGRAPH 78 Conference Proceedings*. 12-19.
10. Francisco Gabriel Ortiz Zamora, Procesamiento morfológico de imágenes en color (2002) Tesis Doctoral. Universidad de Alicante. Alicante, España.
11. Nora La Serna Palomino, Walter Contreras, María Elena Ruiz, Procesamiento Digital de textura: Técnicas utilizadas en aplicaciones actuales de CBIR (2010). *Revista de Investigación de sistemas e Informática*. 7(1), 57-64.
12. Manish H. Bharati, J. Jay Liu, and John F. MacGregor. Image texture analysis: methods and comparisons. *Chemometrics and Intelligent Laboratory Systems* (2004). *Chemometrics and Intelligent Laboratory Systems*, 72, 57-71
13. D. José Orlando Maldonado Bautista (2008). *Estudio de métodos de indexación y recuperación en bases de datos de imágenes*. Tesis Doctoral, Universidad del País Vasco. España.
14. José Luís Gil Rodríguez (2008). Estado Actual de la Representación y Análisis de Textura en Imágenes.
15. Haralick R. M. (1979). Statistical and structural approaches to texture, *Proceedings. IEEE*. 67. 786-804.
16. Haralick, Robert. M., Shanmugam K., Dinstein, Its'Hak: Textural features for image classification. *IEEE Systems, Man, and Cybernetics Society*
17. Galloway, M. M. (1975). Texture analysis using gray level run-lengths. *Computer Graphics and Image Processing*. 4(2), 172-179
18. L. Enrique Sucar, Giovani Gómez. *Visión computacional*. Puebla. México
19. Zhang Lei, Lin Fuzong, Zhang Bo (1999). A CBIR method based on Color-Spatial feature. *TENCON 99, Proceedings of the IEEE Region 10 Conference*. 1, 166-169.
20. Salvador Godoy Calderón . *Evaluación de algoritmos de clasificación basada en el modelo estructural de cubrimientos*. Tesis Doctoral. Centro de investigación en computación del Instituto Politécnico Nacional. México, México.
21. Enrique Bonzón Ponte, Tomás Escobar Rodríguez, M^a del Pilar Martín Zamora. Sistemas de inducción de árboles de decisión: utilidad en el análisis de crisis bancarias, Grupo de Inteligencia Artificial en Contabilidad y Administración de Empresas. Disponible en:
<http://www.ciberconta.unizar.es/biblioteca/0007/arboles.html>
22. S. Y. Kung (1993), *Digital neural networks*. Prentice Hall, Inc. Estados Unidos. Michigan.
23. C. Lau (1991). *Neural Networks, Theoretical Foundations and Analysis*. *IEEE Press*. Estados Unidos. Michigan.
24. Airel Pérez Suárez, José Fco. Martínez Trinidad, José E. Medina Pagola, Jesús Ariel Carrasco Ochoa (2010). Algoritmos dinámicos para el agrupamiento con traslape. Coordinación de Ciencias Computacionales INAOE.

25. D. Pascual, F. Pla, S. Sánchez (2007). Algoritmos de agrupamiento, Universidad de Oriente. Cuba. Cuba.
26. Miguel Garre, Juan José Cuadrado, Miguel Ángel Sicilia (2007), Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software. *Revista Española de Innovación, Calidad e Ingeniería del Software*. 3(1). 6-22.