

PROYECTO DE FIN DE CARRERA

Presentado a

LA UNIVERSIDAD DE LOS ANDES

FACULTAD DE INGENIERÍA

DEPARTAMENTO DE INGENIERÍA
ELÉCTRICA Y ELECTRÓNICA

Para obtener el título de

INGENIERO ELECTRÓNICO

por

Hernán Darío Mejía Mora

*ADQUISICIÓN DE CARACTERÍSTICAS DE FORMA Y RANKING
DE CONTENIDO EN VIDEO USANDO SUPPORT VECTOR
MACHINES Y APRENDIZAJE ACTIVO*

Sustentado el día 30 de Mayo de 2012

Asesor: Fernando Enrique Lozano Martínez, Profesor Asociado, Universidad de Los Andes
Jurados: Reinaldo Augusto Uribe Muriel, Candidato a Doctorado, Universidad de Los Andes

Índice general

1. Introducción	4
2. OBJETIVOS	5
2.1. Objetivo General	5
2.2. Objetivos Específicos	5
2.3. Alcance y productos finales	5
3. MARCO TEÓRICO, CONCEPTUAL E HISTORICO	7
3.1. Marco Teórico y Conceptual	7
3.1.1. Procesamiento de Imágenes	7
3.1.2. Segmentación de Imágenes	7
3.1.3. Evolución discreta de curvas	8
3.1.4. Support Vector Machines	9
3.1.5. Ranking Bipartito	9
3.1.6. SVM-Rank	10
3.1.7. Aprendizaje Activo	10
3.1.8. Ranking Activo Diverso	11
4. DEFINICION Y ESPECIFICACIÓN DEL TRABAJO	13
4.1. Definición	13
5. METODOLOGIA DEL TRABAJO	14
6. Diseño e implementación	16
6.1. Sistema de abstracción de características	16
6.2. Sistema de ranking activo	20
7. VALIDACIÓN DEL TRABAJO	24
7.1. Metodología de prueba	24
7.2. Validación de los resultados del trabajo	24
8. DISCUSIÓN	30
9. CONCLUSIONES	31

Índice de figuras

3.1. Diferentes versiones de una curva evolucionada por el algoritmo de evolución discreta [14]	9
5.1. Diagrama básico del plan de trabajo	15
6.1. Proceso de abstracción de características	17
6.2. Estados del procesamiento de imágenes:(a)Imagen original en escala de grises; (b) Detección de bordes; (c) Dilatación de bordes; (d) Imagen Segmentada	18
6.3. Iteración de la evolución de curvas	20
6.4. Proceso de etiquetado y ranking	21
6.5. Interfaz de Etiquetado de Curvas	22
6.6. Conexión global del sistema final	23
7.1. Diferentes versiones de una curva (OMA): (a) Imágen original;(b) 100 vértices; (c) 50 vértices; (d) 30 vértices; (e) 20 vértices	25
7.2. Diferentes versiones de una curva (Carrefour): (a) Imágen original;(b) 100 vértices; (c) 50 vértices; (d) 30 vértices; (e) 20 vértices	26
7.3. Representación final de varias formas básicas	28
7.4. Error en modo activo(continua) y modo pasivo(punteada), concepto (Carrefour)	29

Capítulo 1

Introducción

En las últimas décadas ha surgido un gran interés en el manejo y procesamiento de bancos de datos para obtener información relevante. El tamaño y complejidad de la información usualmente hace que esta sea imposible de manejar y procesar manualmente. Dentro de estos problemas, el reconocimiento de patrones en imágenes es actualmente un área de estudio [1] y de investigación. Diferentes aproximaciones se han propuesto y logran solucionar de forma específica diversos problemas de reconocimiento de imágenes [2] [3] [4] [5].

Un problema específico ocurre cuando se necesita obtener un ordenamiento de las imágenes dentro de un banco de datos para satisfacer ciertos criterios de relevancia requeridos por una entidad. Tales criterios pueden depender del contexto a trabajar tales como colores [6], texturas [7], formas [4] entre otros, siendo el criterio de forma un elemento de interés debido al potencial de comprensión visual local que brinda este criterio.

El uso de herramientas computacionales y algoritmos automáticos se ha convertido en la solución por preferencia en estas situaciones. Los algoritmos de aprendizaje supervisado tales como Support Vector Machines, Redes Neuronales o Boosting se suelen usar dentro de este contexto debido a su capacidad de solucionar problemas de alta complejidad y a su adaptabilidad para encontrar reglas generales a partir de ejemplos [8] [9] [10]. Este proyecto busca ahondar en la temática de aprendizaje supervisado en el contexto de procesamiento de imágenes para proponer un algoritmo de ranking de videos según criterios de relevancia relacionados con presencia de formas en videos.

Capítulo 2

OBJETIVOS

2.1. Objetivo General

Este proyecto busca desarrollar un sistema computacional que arroje un ordenamiento de relevancia para un banco de contenidos en video según un requerimiento, haciendo uso de procesamiento de imágenes y herramientas de aprendizaje supervisado.

2.2. Objetivos Específicos

- Ahondar en las técnicas de procesamiento de imágenes actuales.
- Implementar un algoritmo de extracción y representación de características de forma en videos aptas para su uso en un sistema de aprendizaje supervisado.
- Recolectar de forma exitosa uno o más bancos de datos de video para el proceso de entrenamiento y prueba, siendo estos lo suficientemente diversificados en cuanto a conceptos y duración para un adecuado ejercicio de entrenamiento y prueba.
- Adaptar las técnicas de Aprendizaje Activo actuales al problema de ranking de contenido.
- Plantear e implementar un algoritmo de ranking de contenido en videos haciendo uso de ranking con Support Vector Machines y Aprendizaje Activo.

2.3. Alcance y productos finales

El alcance de este proyecto es el de implementación exitosa de un sistema que soluciona la problemática de ranking en video, no necesariamente solucionando el problema para el uso de un usuario final, sino una solución investigativa como primer acercamiento a esta temática. Para esto se tienen los siguientes productos finales:

- El presente documento
- Código fuente en MATLAB para el sistema de abstracción de características en video, sistema de etiquetado gráfico y sistema de ranking.
- Archivos de modelo para ranking de los conceptos aprendidos en el proyecto.

Capítulo 3

MARCO TEÓRICO, CONCEPTUAL E HISTORICO

3.1. Marco Teórico y Conceptual

3.1.1. Procesamiento de Imágenes

EL procesamiento de imágenes es una amplia disciplina compuesta de todo tipo de técnicas de tratamiento y análisis de señales que buscan brindar algún tipo de información en imágenes. Dentro de estas técnicas se encuentran aquellas denominadas de preprocesamiento que buscan acondicionar las imágenes para procesamientos superiores, técnicas de transformación, que se usan para representar la información en diferentes espacios, técnicas de procesamiento morfológico tales como de detección de bordes y segmentación de imágenes se usan en problemas de visión artificial y reconocimiento de patrones, entre otras [11].

3.1.2. Segmentación de Imágenes

En diferentes problemas relacionados con procesamiento de imágenes se hace necesario convertir una imagen en diferentes segmentos que satisfacen características en común. Diferentes enfoques tales como de detección de borde, transformada Hough para detección de rectas, crecimiento de regiones y transformada “Watershed” satisfacen requerimientos para diferentes criterios de segmentación.

El enfoque de segmentación por detección de bordes es de amplio uso debido a sus diversas aplicaciones y a su nivel de generalización. A su vez, la detección de bordes dispone de diferentes aproximaciones o algoritmos, tal como los detectores Sobel, Prewitt, Roberts, Laplaciano de una Gaussiana (LoG), cruces por cero y finalmente el algoritmo de Canny. Este último es considerado superior a los otros debido a su habilidad diferenciar claramente bordes falsos positivos y verdaderos positivos. La segmentación entonces consiste en separar las regiones que se encuentran dentro de un contorno de borde ce-

rado para formar una región independiente de las otras en el espacio de la imagen.

3.1.3. Evolución discreta de curvas

La evolución discreta de curvas surge de la necesidad de expresar una curva poligonal cerrada en versiones poligonales de menor complejidad y menor cantidad de vértices [12][13], esto con el fin de reducir ruido de borde y segmentación, suavizar una curva y enfatizar los rasgos más característicos de la forma que se intenta representar con la curva. El algoritmo entonces toma un polígono cerrado y realiza un proceso iterativo de eliminación de vértices. En este proceso se calcula un coeficiente de relevancia de vértices definido de la siguiente forma

$$K_i = \frac{(\theta_i)(l_{i-1,i}l_{i,i+1})}{l_{i,i+1} + l_{i-1,i}} \quad (3.1)$$

Donde θ_i es el ángulo de giro entre los dos aristas del vértice i y además $0 \leq \theta_i \leq \pi$, y $l_{i,j}$ es la longitud entre los vértices i y j . A partir de la normalización de una curva para que su perímetro sea unitario, el coeficiente de relevancia K_i toma la información del vértice i y los vértices adyacentes $i+1$ e $i-1$ e indica qué tan importante es un vértice para describir la forma de en su totalidad. Así en cada iteración de la evolución discreta de curvas se realiza una normalización de la curva para que su perímetro sea 1, para cada vértice se halla la longitud de sus aristas y el ángulo interno formado por ellas, el menor coeficiente de relevancia entre todos los vértices es eliminado y el nuevo polígono es formado por los vértices restantes. La evolución discreta de curvas busca encontrar diferentes versiones de una misma curva en la cual cada versión enfatiza diferentes niveles de información y detalle. Usualmente las primeras iteraciones del algoritmo de evolución de curvas eliminan el ruido de la imagen y los vértices que no representan información de la imagen, posteriormente se suaviza la curva para obtener eventualmente una curva que contiene algunos de los vértices originales de la imagen pero que son los más relevantes de visualmente para describir la curva. El factor de relevancia K está diseñado para darle prioridad a los vértices más enfáticos de una imagen, por lo que da prioridad a ángulos pronunciados (convexos o cóncavos), a vértices cuya longitud de sus aristas es similar y a aristas cuya longitud es mayor. La figura 3.1 muestra diferentes versiones de una misma curva de la cual se observa que ocurre eliminación de ruido, caracterización de la figura por aspectos relevantes y finalmente convergencia a una curva convexa con algunos de los vértices originales.

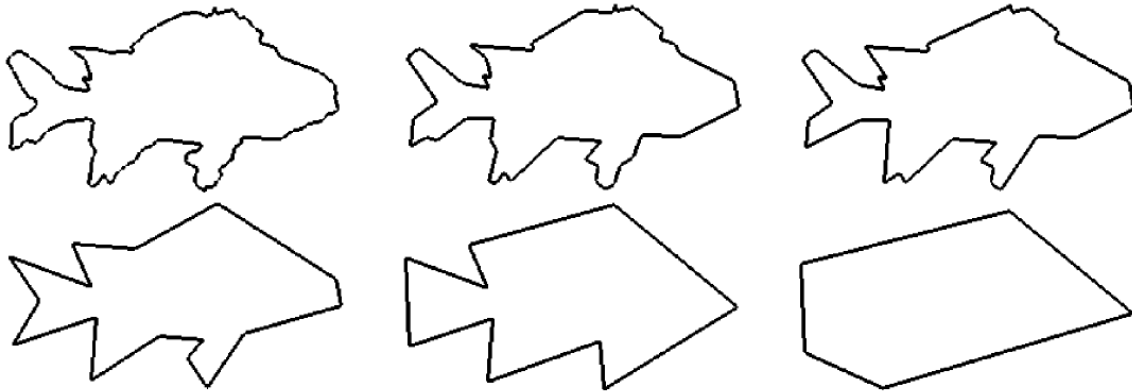


Figura 3.1: Diferentes versiones de una curva evolucionada por el algoritmo de evolución discreta [14]

3.1.4. Support Vector Machines

Los Support Vector Machines [15] son sistemas de aprendizaje de máquinas que buscan encontrar un separador lineal que maximice el margen entre las clases que se separan. Para esto, se busca solucionar el siguiente problema de optimización

$$\begin{aligned}
 & \text{mín } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i \\
 & \text{s.a } y_i (\langle w, x_i \rangle + b) - 1 + \xi_i \geq 0 \\
 & \quad i = 1, 2, \dots, N \\
 & \quad \xi_i \geq 0
 \end{aligned} \tag{3.2}$$

Donde ξ_i son variables de holgura usadas para manejar la no separabilidad de clases, C es un parámetro de regularización que indica un balance entre error y margen. De igual manera, en SVM se hace uso de Kernels los cuales se usan para solucionar el problema en casos no lineales al llevar los datos a un espacio de alta dimensionalidad denominado *Espacio de Hilbert*; la potencia de los kernels permite gran flexibilidad en las distribuciones de entrada de los datos permitiendo solucionar gran variedad de problemas de aprendizaje supervisado y reconocimiento arrojando buenos resultados[16] [8] [9] por lo que actualmente es una de las herramientas de aprendizaje supervisado más usadas. El problema de optimización a solucionar también requiere del ajuste de los diferentes parámetros del Kernel usado y el parámetro C y suele ser un problema a resolver en diferentes sistemas de aprendizaje [17] [18].

3.1.5. Ranking Bipartito

Los problemas de ranking bipartito [19] consisten en buscar un ordenamiento considerando que existen dos clases de datos: relevantes y no relevantes. En este caso no se

busca encontrar un clasificador que asigne pertenencias a una u otra clase de conceptos, sino se busca que los ejemplos relevantes sean puntuados por encima de los ejemplos negativos por lo que mientras más ejemplos negativos están por encima de un ejemplo positivo, mayor error ocurre, lo que conlleva a la definición de *error de ranking empírico*

$$\hat{e}r_S(f) = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n \mathbf{1}(f(x_i^+) < f(x_j^-)) \quad (3.3)$$

La cual es una medida de desempeño para una función de ranking f usando un conjunto de datos S .

3.1.6. SVM-Rank

El algoritmo de SVM-Rank fue propuesto en [20] en el cual se propone usar parejas de ejemplos (x_i, x_j) y asignar un ordenamiento de relevancia individual a cada par de ejemplos. El problema de optimización a solucionar es similar al problema original de SVM en clasificación binaria; en este caso se busca encontrar el vector de pesos w que maximice el margen entre comparaciones. Al tratar comparaciones de datos como ejemplos para el problema de optimización, el problema se puede solucionar como si fuera un problema convencional de clasificación binaria.

3.1.7. Aprendizaje Activo

En el contexto de aprendizaje activo se considera que la información de pertenencia de clases o etiquetas de los datos es accesible pero a un costo, por lo cual es importante acceder a la menor cantidad de etiquetas de forma que se minimice el costo de obtención e igualmente se maximice el desempeño del clasificador. De esta manera en aprendizaje activo se busca tener el mejor desempeño posible de un clasificador para la menor cantidad de datos posible.

Los algoritmos de aprendizaje activo buscan entonces a partir de un clasificador base construido a partir de ejemplos base encontrar los ejemplos que brindan más información a un modelo para así mejorar el rendimiento lo mejor posible [21]. Entre diferentes enfoques de aprendizaje activo, se encuentra el enfoque basado en selección de datos dentro de un conjunto de candidatos sin etiquetar, esta situación ocurre cuando no es posible seleccionar el ejemplo óptimo a etiquetar dentro de una distribución de entrada, sino que se dispone únicamente de un conjunto de ejemplos y se debe buscar el ejemplo que brinda mayor información. Las técnicas de aprendizaje activo han sido aplicadas en el contexto de procesamiento de señales e imágenes [22] [23], [24] donde en el contexto de clasificación brindan mejoras en los tiempos de aprendizaje de un concepto.

Uno de los principales problemas del aprendizaje activo es denominado "sesgo por muestreo" [25] el cual surge debido a que observar los datos más informativos de una distribución de entrada produce un sesgo en la estimación de la distribución lo que conlleva a errores de clasificación sistemáticos, el problema entonces consiste en encontrar

un balance entre exploración de los datos y explotación de los datos más informativos. Es por esto que al abordar técnicas de aprendizaje activo se debe tener en cuenta este sesgo y se debe intentar minimizar lo más posible dependiendo del problema.

3.1.8. Ranking Activo Diverso

Aunque los problemas de clasificación mediante aprendizaje activo han sido ampliamente tratados, no ocurre lo mismo con el aprendizaje activo enfocado a ranking donde hay pocas soluciones dependientes del problema [26]. En este caso no hay un hiperplano separador que divida las clases, solamente comparaciones entre pares que indican relevancia, y como resultado final se obtiene una función $f \in \mathbb{R}$ cuyo puntaje permite conocer la relevancia de un ejemplo. La aproximación de aprendizaje activo en clasificación buscaría etiquetar los ejemplos en los cuales es más difícil o más confuso clasificar un ejemplo, pero en [27] se sugiere que con pequeñas cantidades de datos es mejor seleccionar un grupo de ejemplos *diversos* dentro de la distribución de entrada en lugar de ejemplos más confusos para el sistema a entrenar debido a que datos muy similares pueden pertenecer a la misma clase y no se pueden comparar en un ranking.

Diversidad La diversidad es una que intenta medir la aleatoriedad de una variable, existen diferentes métodos de estimación de la diversidad de un conjunto de datos. La estimación de la diversidad mediante la entropía consiste en estimar la entropía para un set de datos y relacionarla con la diversidad del conjunto. Para un set de datos de tamaño N , la entropía de un conjunto se puede aproximar por la siguiente ecuación.

$$h(x) \approx -\frac{1}{N} \sum_{j=1}^N \log(p(x_j)) \quad (3.4)$$

Debido a su aproximación probabilística se hace uso de ventanas de Parzen [28] las cuales se usan como estimadores de probabilidad mediante la siguiente ecuación,

$$p(x) = \frac{1}{N} \sum_{i=1}^N W(x, x_i) \quad (3.5)$$

Dentro de las diferentes ventanas de Parzen, la ventana Gaussiana se suele usar en este método la cual está definida de la siguiente manera,

$$W(x, x_n) = \frac{1}{(2\pi h^2)^{d/2}} e^{-\frac{\|x-x_n\|^2}{2h^2}} \quad (3.6)$$

Índice de claridad El índice de claridad es una estimador de cuan fácil es aplicarle un ranking a un ejemplo no etiquetado, es decir, indica la seguridad que tiene un sistema de ranking de que el ejemplo pertenezca a una clase. Así, sea $T = ((x_1, y_1), \dots, (x_N, y_N))$

el conjunto de los datos etiquetados en una iteración dada, sea x_i^u un dato dentro del conjunto de datos no etiquetados y sea f la función obtenida en la iteración actual del sistema de ranking con el conjunto T . Sea entonces la *pérdida de relevancia* definida de la siguiente manera

$$RL(x_i^u, f, T) = \frac{1}{n_0} \sum_{j=1}^n \mathbf{1}(f(x_i^u) < f(x_j^-)) \quad (3.7)$$

y la *pérdida de irrelevancia* definida de la siguiente manera

$$IL(x_i^u, f, T) = \frac{1}{n_1} \sum_{j=1}^n \mathbf{1}(f(x_i^u) > f(x_j^+)) \quad (3.8)$$

donde x^-, n_0 y x^+, n_1 son los ejemplos negativos y positivos respectivamente dentro del conjunto T y sus tamaños en el conjunto. La *pérdida de relevancia* puede ser interpretada como el error de ranking de un conjunto compuesto por $((x_i^u, 1), x^+)$ e igualmente la *pérdida de irrelevancia* puede ser interpretada como el error de ranking del conjunto $((x_i^u, 0), x^-)$.

Finalmente se define el *Índice de claridad* de un ejemplo no etiquetado x_i^u así,

$$CI(x_i^u, f, T) = |RL(x_i^u, f, T) - IL(x_i^u, f, T)| \quad (3.9)$$

Selección de ejemplos El método de selección de ejemplos dentro de un conjunto de datos no etiquetados candidatos C consiste entonces en el siguiente: Se inicializa el conjunto QS el elemento que tiene el menor índice de claridad dentro del conjunto C . Posteriormente se hace una selección de manera codiciosa en K iteraciones de los puntos que al agregarlos al conjunto QS maximizan la entropía o diversidad en cada iteración del conjunto $C \setminus QS$

Capítulo 4

DEFINICION Y ESPECIFICACIÓN DEL TRABAJO

4.1. Definición

Partiendo de la problemática planteada, este proyecto consiste en el desarrollo e implementación de un sistema computacional práctico que entregue un puntaje de ranking para diferentes conceptos aprendidos a elementos de un banco de videos.

Inicialmente se busca que el sistema sea aplicable a conceptos de la vida real según las limitaciones que genere el sistema en su desarrollo.

El trabajo está encaminado a una solución investigativa al problema de ranking de contenido en videos y busca aplicar un método de solución en base a diferentes autores y a una propuesta propia de representación de características. Por esta razón el trabajo aplicado en el proyecto se basa en una investigación exhaustiva sobre el estado del arte de este tema para llegar a la implementación de un algoritmo global.

Capítulo 5

METODOLOGIA DEL TRABAJO

El proyecto fue estructurado para realizar una implementación de un sistema que soluciona el problema descrito. De esta manera se planteó el diseño de módulos independientes que realicen un único proceso para después conectarlos y conformar el sistema final.

El plan de trabajo estuvo entonces dividido en los siguientes módulos:

- Etapa de investigación y recolección de datos, en la cual se realizó una exhaustiva revisión bibliográfica así como recolección de datos reales para usar en este proyecto.
- Etapa de implementación del algoritmo de abstracción de formas.
- Desarrollo del sistema de ranking activo.

Aunque las etapas son independientes la figura 5.1 muestra la conectividad seguida en el desarrollo del proyecto, un proceso iterativo en el cual se realiza realimentación a las etapas para obtener implementaciones válidas según los requerimientos de cada etapa.

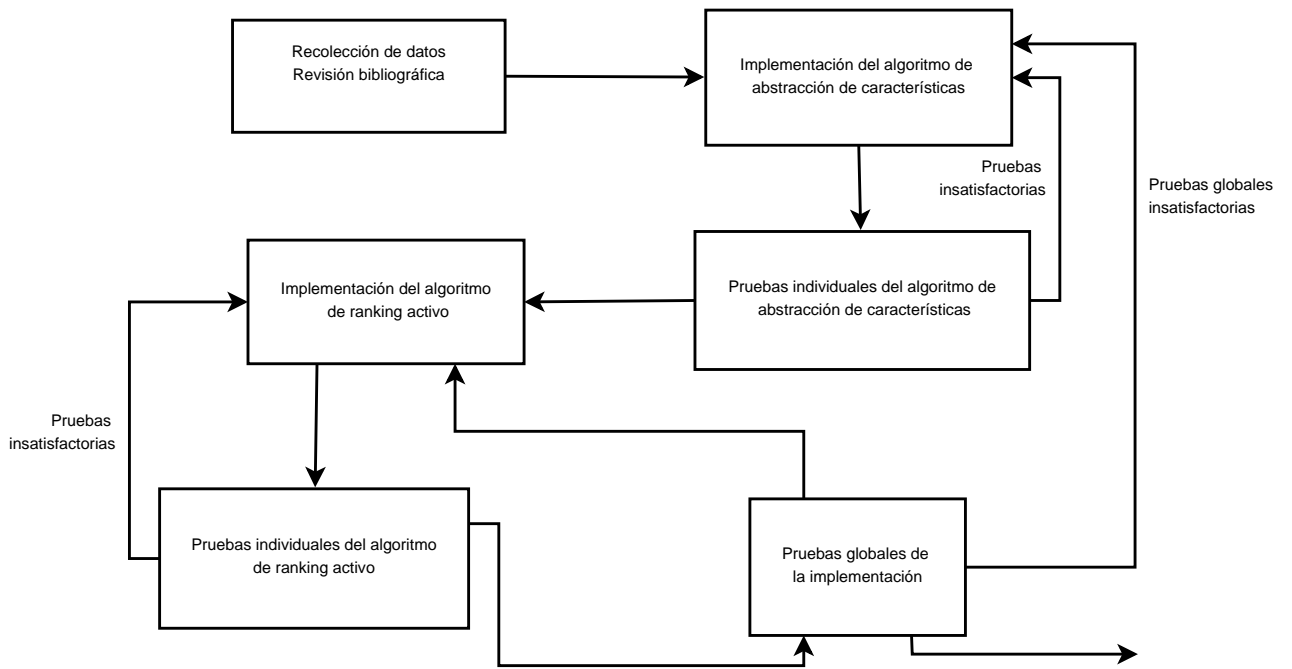


Figura 5.1: Diagrama básico del plan de trabajo

Capítulo 6

Diseño e implementación

La implementación del proyecto fue netamente basada en el entorno de programación de MATLAB® donde se implementaron diversas funciones que cumplieran los requerimientos individuales.

6.1. Sistema de abstracción de características

El sistema de abstracción de características de forma fue implementado teniendo en cuenta que se debían procesar las imágenes de los videos de forma individual y en cada una de ellas se realizaría el proceso de representación. En este proceso, los videos son descompuestos en diferentes niveles de representación para lograr un procesamiento a bajo nivel, posteriormente se agrupa la información procesada a bajo nivel en un contenedor de alto nivel. El proceso seguido se describe en la figura 6.1, tomando como entrada el conjunto de videos y realizando la descomposición descrita a continuación.

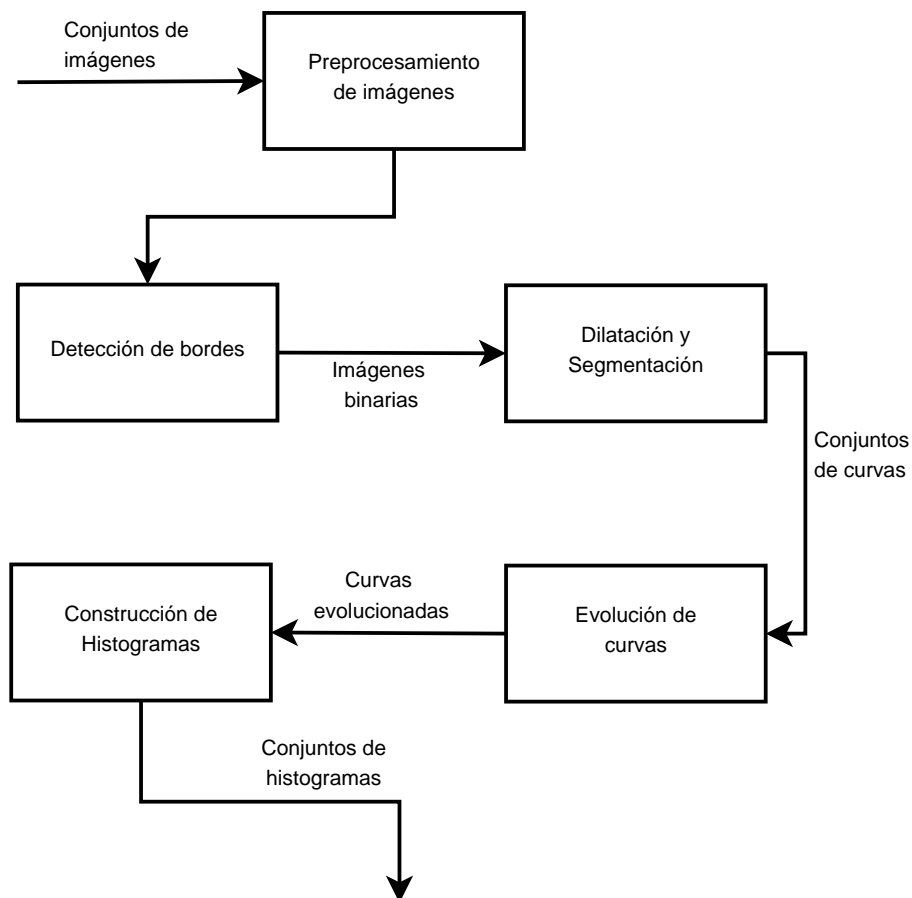


Figura 6.1: Proceso de abstracción de características

Preprocesamiento de imágenes Los videos en principio se consideraron como contenedores de imágenes a las cuales se les aplicó un preprocesamiento básico: transformación a imágenes en escala de grises y procesamiento lineal de realce de contraste con el fin de facilitar la detección de bordes en las etapas superiores.

Detección de bordes Se usó el algoritmo de detección de bordes Canny [29], el cual hace uso de filtrado gaussiano para mitigar los efectos de ruido y posteriormente aplica el operador de detección de bordes Sobel de forma vertical, horizontal, diagonal positiva y negativa que busca calcular el gradiente bidimensional de la imagen y aplicar un umbral para detectar los bordes, este umbral es dependiente de la imagen y es ajustado automáticamente. El resultado es una imagen binaria indicando las coordenadas donde hay presencia de bordes.

Dilatación y Segmentación Debido a que las imágenes binarias generadas por la detección de bordes no agrupan los bordes como segmentos continuos se debe realizar un procesamiento morfológico en el cual los pixeles de borde son dilatados para lograr

conectividad de segmentos de bordes. La dilatación aplica un barrido de la imagen a la cual se le aplica una estructura morfológica en forma de diamante de tamaño fijo de 3 píxeles de ancho y alto. De esta manera los segmentos de borde tienen un ancho definido y se consideran entonces como barreras entre formas. La imagen puede entonces ser considerada como un conjunto de formas separadas por barreras que representan los bordes. A partir de esto, la segmentación de la imagen consiste en representar por medio de los píxeles de borde al conjunto de formas que componen la imagen. La imagen es entonces expresada como un contenedor de formas las cuales tienen longitudes y tamaños variables. La imagen 6.2 presenta los diferentes estados de la imagen hasta llegar a la segmentación de formas; para facilitar la representación visual las formas segmentadas son separadas por colores.

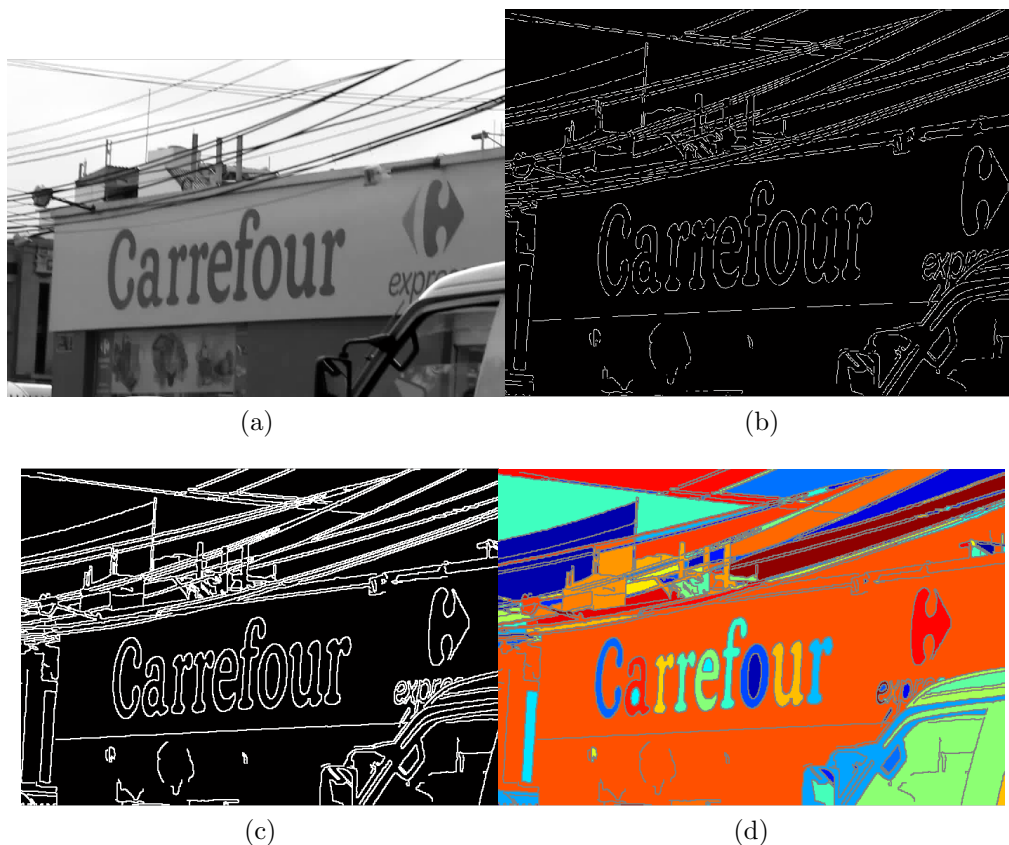


Figura 6.2: Estados del procesamiento de imágenes:(a)Imagen original en escala de grises; (b) Detección de bordes; (c) Dilatación de bordes; (d) Imagen Segmentada

Evolución de Curvas Las curvas obtenidas dentro de las imágenes son entonces evolucionadas por medio del algoritmo de evolución de curvas, por conveniencia algorítmica se plantea un factor de relevancia K que depende del ángulo interno formado por los

rayos de cada vértice, así el factor de relevancia usado es,

$$K_i = \frac{|\pi - \beta_i|(l_{i-1,i}l_{i,i+1})}{l_{i,i+1} + l_{i-1,i}} \quad (6.1)$$

Esta definición es equivalente y no modifica la relevancia de un vértice, aunque debido a que $0 \leq \beta \leq 2\pi$ es posible diferenciar entre ángulos internos agudos y obtusos, esta diferenciación es de gran importancia ya que ángulos de giro equivalentes pero opuestos, es decir agudos y obtusos, aportan información de forma totalmente distinta.

El algoritmo implementado inicialmente calculaba todos los coeficientes de relevancia K de una curva en cada iteración el cual era un algoritmo ineficiente, la segunda versión del algoritmo consiste en calcular únicamente el coeficiente K para los vértices que cambian su ángulo debido a la eliminación de un vértice, y reescalar los coeficientes de los demás vértices. Sean los vértices i, j, k en la figura 6.3 donde j es el vértice a eliminar en la iteración. Sea la disminución en el perímetro de la curva definido de la siguiente manera,

$$\Delta L = l_{i,j} + l_{j,k} - l_{i,k} \quad (6.2)$$

Debido a que la curva generada en la siguiente iteración debe ser nuevamente normalizada para que su perímetro sea unitario, este cambio implica multiplicar por un factor de escalamiento $\frac{1}{1 - \Delta L}$ por lo que cada segmento de curva será reescalado por esta misma proporción. Finalmente el valor de K en la iteración n queda así,

$$K_m(n) = \begin{cases} \frac{|\pi - \beta_i|(l_{m-1,m}l_{m,m+1})}{l_{m,m+1} + l_{m-1,m}} & \text{para } m=j,k \\ \frac{K_m(n-1)}{1 - \Delta L} & \text{d.l.c} \end{cases} \quad (6.3)$$

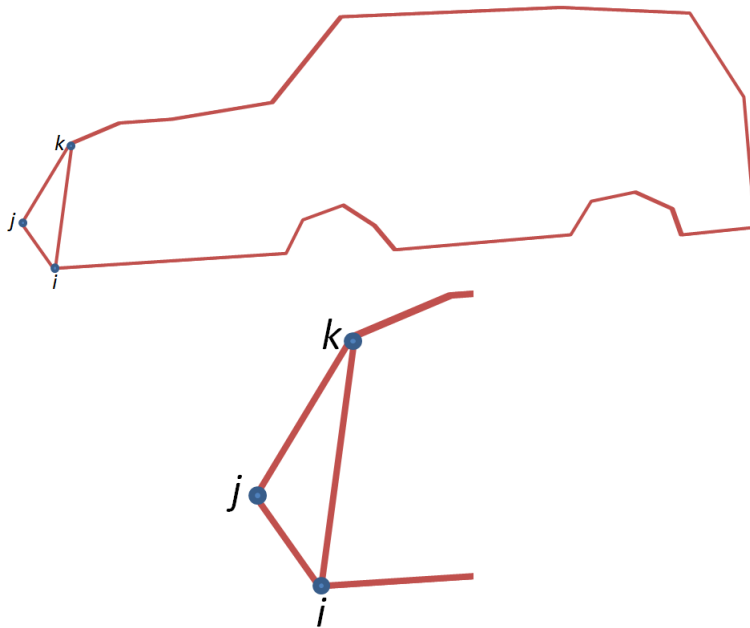


Figura 6.3: Iteración de la evolución de curvas

Histogramas de curvas Debido a que se buscaba un descriptor de formas independiente a rotación y tamaño de una curva, y además un descriptor de tamaño fijo. Se planteó un descriptor basado en histogramas angulares y de longitud debido a que una curva se puede considerar como una distribución única de ángulos y longitudes normalizadas. Así la curva es evolucionada hasta cierto punto, tomando diferentes versiones de la curva en el proceso. Para cada versión de la curva se construyó un histograma angular y de longitud, dando prioridad al histograma angular en la cantidad elementos del descriptor. Dado que un histograma agrupa los elementos que están dentro de cierto rango, hay pérdida de especificidad al no poder diferenciar entre el valor de un elemento u otro dentro de un mismo rango. Para esto se diseñó un histograma con ventanas variables: sea el rango de una clase $B := [a, b)$ entonces el aporte de un dato $x \in B$ a la clase es

$$s(x) = \frac{0,5(x - a)}{b - a} + 0,5 \quad (6.4)$$

Lo cual significa que un dato perteneciente a un rango aportará un valor único a su clase entre 0.5 y 1 al histograma. Este método brinda una mayor especificidad en la representación ya que elimina la discretización que genera la construcción de un histograma convencional.

6.2. Sistema de ranking activo

Dadas las características de una curva representadas según la sección anterior es posible plantear un problema de ranking bipartito en el cual se desea que las curvas

relevantes pertenecientes a un concepto sean ordenadas en una relevancia más alta que las curvas irrelevantes según una función de ranking $f(x)$. Suponiendo que una imagen se hace relevante ante la presencia de una curva relevante en su conjunto de curvas y que se conoce qué curva hace relevante la imagen por medio de etiquetas de relevancia o irrelevancia, el problema se puede manejar a nivel de ranking de curvas para llegar a un ranking de imágenes y así a un puntaje de videos para ordenamiento. El sistema implementado aplica un proceso iterativo de etiquetado automático y entrenamiento. La figura 6.4 representa el proceso de obtención de una función de ranking a partir de la representación de uno o varios videos como contenedores de curvas.

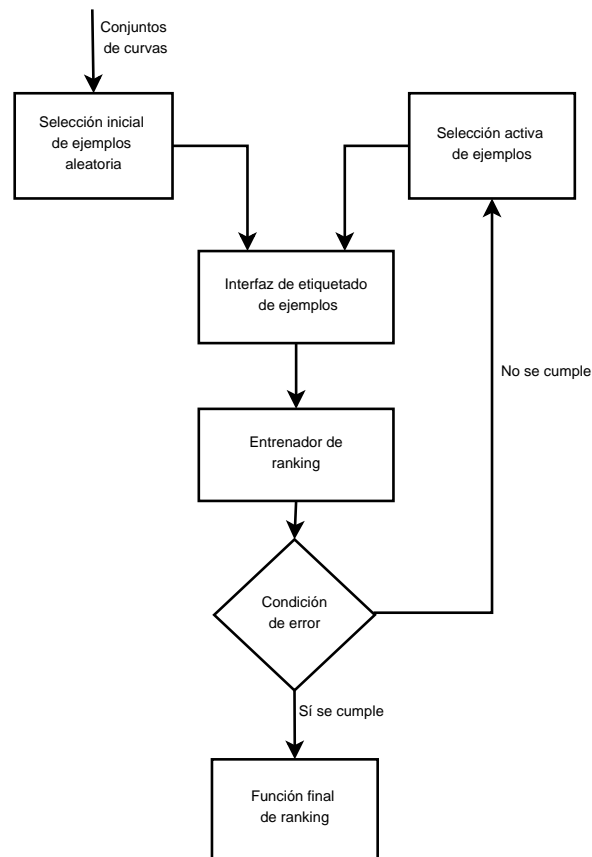


Figura 6.4: Proceso de etiquetado y ranking

Interfaz de etiquetado La interfaz de etiquetado implementada se muestra en la figura 6.5. Esta interfaz está diseñada para realizar un etiquetado por pares de elementos de curvas dentro de dos imágenes. Dado un par de imágenes la interfaz muestra sus representaciones obtenidas por medio de la segmentación de imágenes aplicando color a cada forma de forma aleatoria para facilitar la comprensión visual de la imagen. El usuario entonces procede a hacer clic en una región coloreada dentro de alguna de las dos figuras y que representa el concepto a aprender, la interfaz detecta la forma indicada por el usuario y relaciona su relevancia con su representación por histogramas; los

pares (relevante-irrelevante) son obtenidos tomando la forma relevante fija indicada por el usuario y las formas de la otra imagen como las formas irrelevantes por lo que una iteración del proceso generará tantas parejas como curvas tenga la imagen irrelevante. En caso de que se desee cambiar una u otra imagen, un botón dispuesto con este propósito para cada imagen tomará otra imagen dentro del conjunto de imágenes candidatas al etiquetado, finalmente si se considera que las dos imágenes tienen la misma relevancia y no se desea cambiar a alguna dentro del conjunto de candidatas entonces un botón central ignorará las dos imágenes y pasará a la siguiente iteración de etiquetado hasta completar N iteraciones.

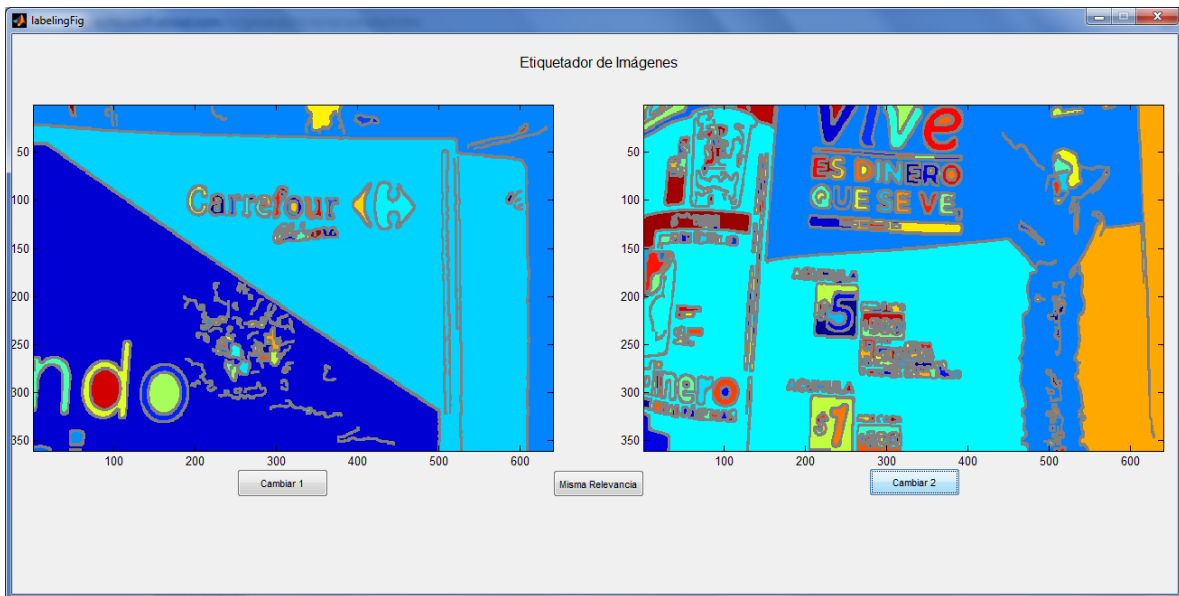


Figura 6.5: Interfaz de Etiquetado de Curvas

Selección de ejemplos Inicialmente se seleccionan los ejemplos de forma aleatoria dentro de los conjuntos de imágenes ofrecidas por el usuario para generar 20 iteraciones de etiquetado y construir una función inicial de ranking.

La selección activa de ejemplos consiste en seleccionar los ejemplos a etiquetar que brindan mayor información. Debido a que los ejemplos que se consideraron fueron pares de curvas en lugar de pares de imágenes, la selección activa de imágenes se hizo a partir de la curva más representativa de cada imagen según el puntaje de ranking asignado por la función de ranking de la iteración anterior, lo cual significa que una imagen puede ser mapeada a una única curva que representa el concepto a aprender. De esta manera se toman todas las imágenes ingresadas por el usuario para entrenar y se encuentra la curva más representativa de cada imagen. A continuación se calcula el índice de claridad CI de cada imagen y se selecciona aquella que tiene el menor índice de claridad, la cual inicializa el conjunto de imágenes candidatas a etiquetar. Posteriormente se seleccionan otras 19 curvas que de forma codiciosa maximizan la entropía del conjunto de curvas representativas siguiendo el método presentado en [27].

Entrenamiento Se hizo uso del paquete SVM-Light en su modo de aprendizaje de funciones de ranking siguiendo el formato de datos apropiado para representar los pares de datos. La configuración usada hace uso de kernel Gaussiano para el cual es necesario encontrar el parámetro γ junto con el parámetro de margen suave C . En [30] se propone un proceso de selección de parámetros mediante búsqueda aleatoria generando usualmente mejores resultados en comparación a la búsqueda de grilla. De esta manera se generan los parámetros C y γ de forma aleatoria dentro de un rango definido para cada variable; para verificar los resultados se hace validación cruzada de 5 iteraciones sobre el conjunto de datos. El proceso de entrenamiento se repite por varias iteraciones al agregar más ejemplos por iteración y reentrenar hasta que se llegue a un criterio de error determinado.

Ranking El puntaje de ranking de un video es obtenido al promediar los puntajes de ranking de sus imágenes al cuadrado, así un video ν con N fotogramas tendrá el siguiente puntaje de ranking:

$$R(\nu) = \sum_{i=1}^N \frac{f(x_i)^2}{N} \quad (6.5)$$

El puntaje se diseñó con el fin de dar mayor relevancia a videos con algunos fotogramas muy relevantes y menor relevancia a videos con fotogramas de relevancia media.

Sistema global El sistema global se describe en la figura 6.6, debido a que se desea independencia entre módulos y una eficiente gestión de memoria, los módulos siempre funcionan a partir de la memoria secundaria de donde se obtienen y guardan los datos según el caso.

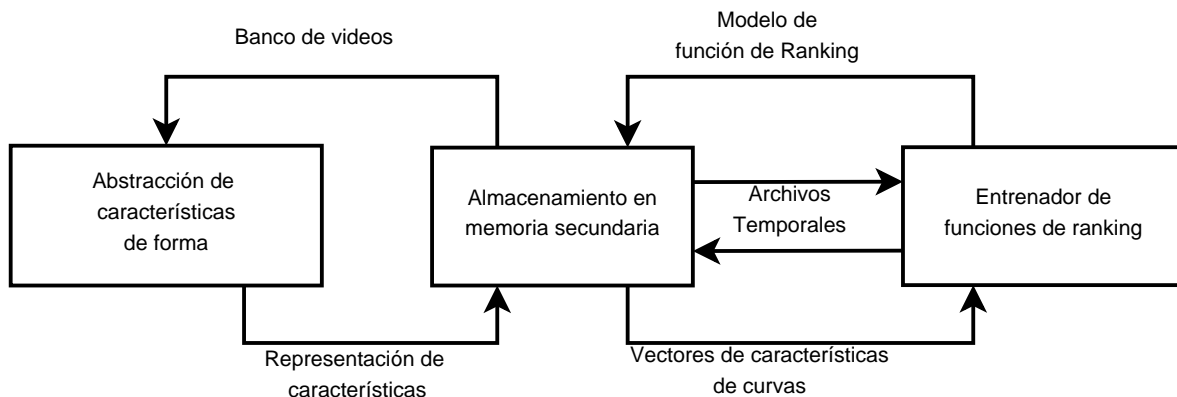


Figura 6.6: Conexión global del sistema final

Capítulo 7

VALIDACIÓN DEL TRABAJO

7.1. Metodología de prueba

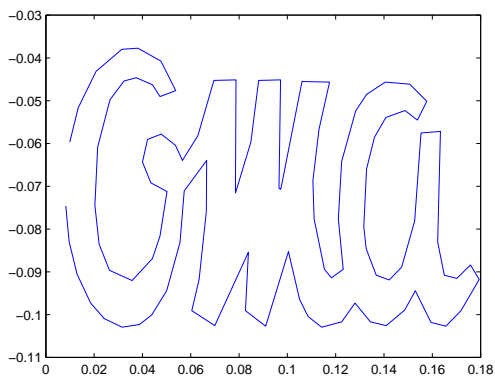
Los sistemas implementados se probaron inicialmente de forma independiente para los cuales se buscaba obtener resultados locales tales como una representación de curvas o una función de ranking. De esta forma fue posible llegar a un funcionamiento independiente de cada módulo; por supuesto para practicar pruebas en el módulo de entrenamiento era necesaria una representación básica de las formas para trabajar con ellas.

7.2. Validación de los resultados del trabajo

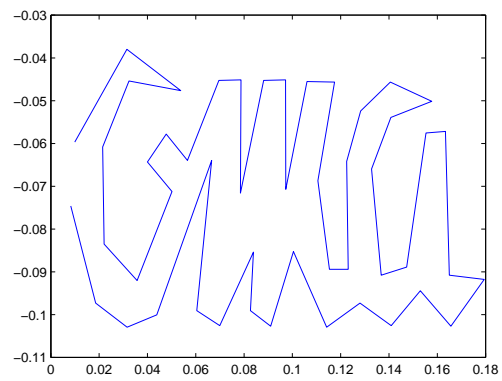
Pruebas individuales del sistema de abstracción de características En este caso se hicieron pruebas para observar cuales eran los resultados de la evolución de curvas para diferentes cantidades de vértices con el fin de encontrar un tamaño fijo que incluyera la menor cantidad de vértices irrelevantes.



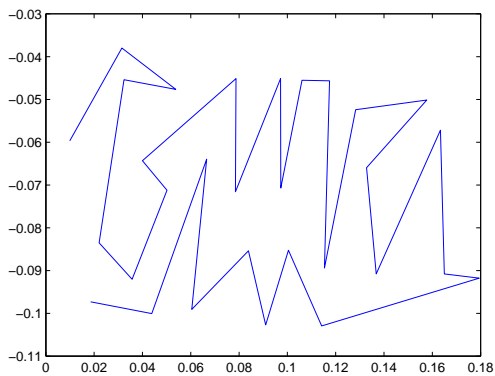
(a)



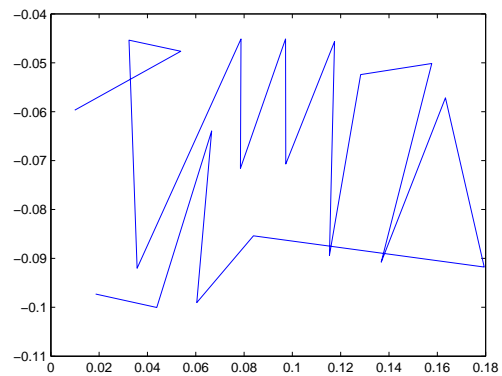
(b)



(c)



(d)

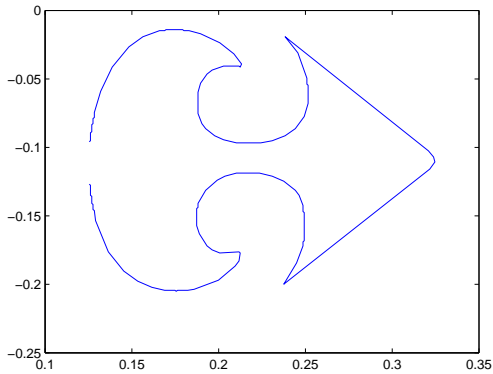


(e)

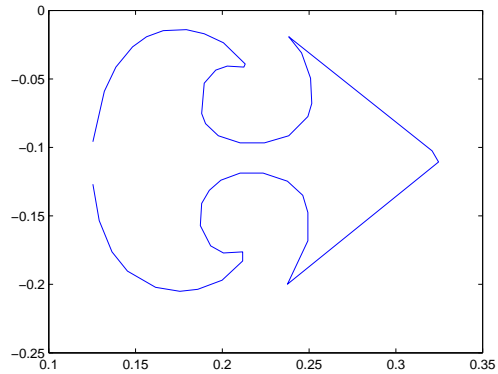
Figura 7.1: Diferentes versiones de una curva (OMA): (a) Imágen original; (b) 100 vértices; (c) 50 vértices; (d) 30 vértices; (e) 20 vértices



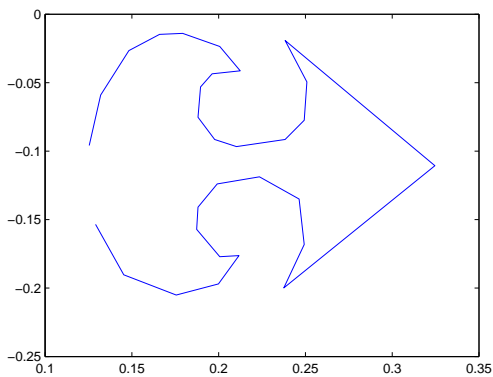
(a)



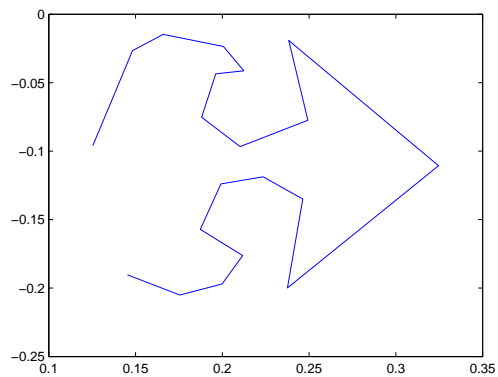
(b)



(c)



(d)

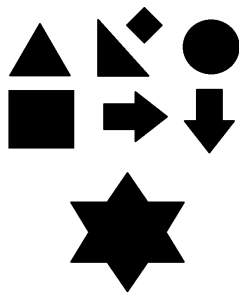


(e)

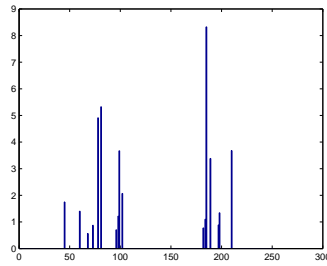
Figura 7.2: Diferentes versiones de una curva (Carrefour): (a) Imágen original;(b) 100 vértices; (c) 50 vértices; (d) 30 vértices; (e) 20 vértices

Para la representación final se escogió una única representación de 30 vértices para representar la curvas debido a que se observó logra el mejor equilibrio entre informatividad para diferentes curvas.

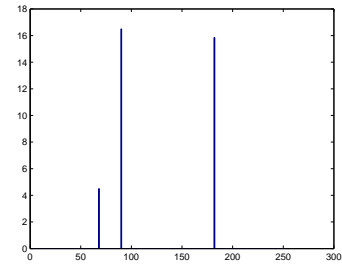
Representación por histogramas Se usaron varias formas de prueba básicas cuyas representaciones sean interpretables para corroborar el funcionamiento del algoritmo la figura 7.3 muestra representaciones para curvas con 30 vértices. Los histogramas disponen de 180 clases iguales para la distribución angular y 70 clases para la distribución de longitudes normalizadas formando un total de 250 clases. El análisis de estas figuras permitió confirmar el funcionamiento del algoritmo de representación de curvas: se pudo observar la similitud entre las representaciones de las dos flechas, lo que confirmó el funcionamiento de la invarianza en rotación buscada; el cuadrado constaba de 1 bin en los 90 grados representando los vértices del cuadrado, otro bin en los 180 grados más grande representando los segmentos de curva en línea recta que forman los bordes, y un bin de longitud representando la longitud de todos los segmentos de la curva evolucionada; el círculo contiene una distribución angular centralizada en un ángulo menor a los 180 grados indicando la curvatura de los segmentos en el estado de evolución propuesto de 30 vértices, La estrella muestra la presencia sistemática de un conjunto de ángulos agudos en la curva y otro de ángulos obtusos en la curva, por lo que la diferenciación se logra satisfactoriamente.



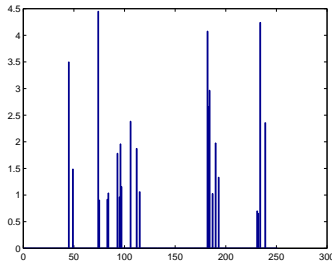
(a) Imagen Original



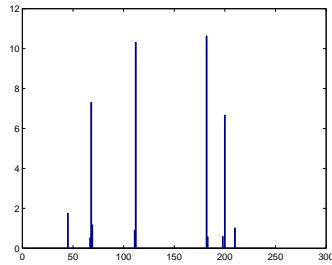
(b) Triángulo



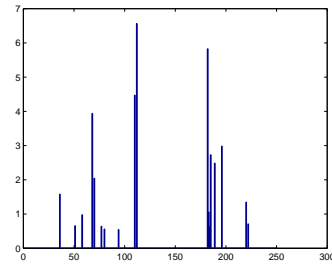
(c) Cuadrado



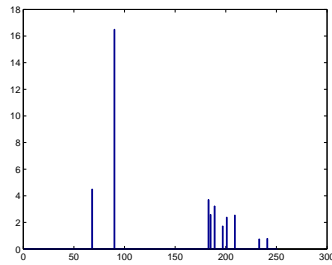
(d) Estrella



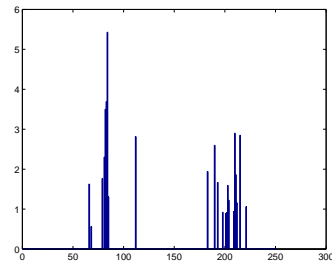
(e) Triángulo rectángulo



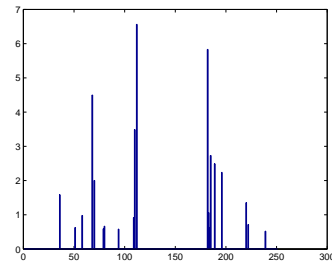
(f) Flecha a la derecha



(g) Rombo



(h) Círculo



(i) Flecha hacia abajo

Figura 7.3: Representación final de varias formas básicas

Ranking Activo vs Ranking Pasivo El entrenamiento de diferentes conceptos mostró una ligera mejoría en la mayoría de los conceptos aprendidos, la figura 7.4 muestra el error obtenido por validación cruzada en las diferentes iteraciones de selección de ejemplos observando una clara mejoría entre la selección activa por diversidad y la selección pasiva aleatoria.

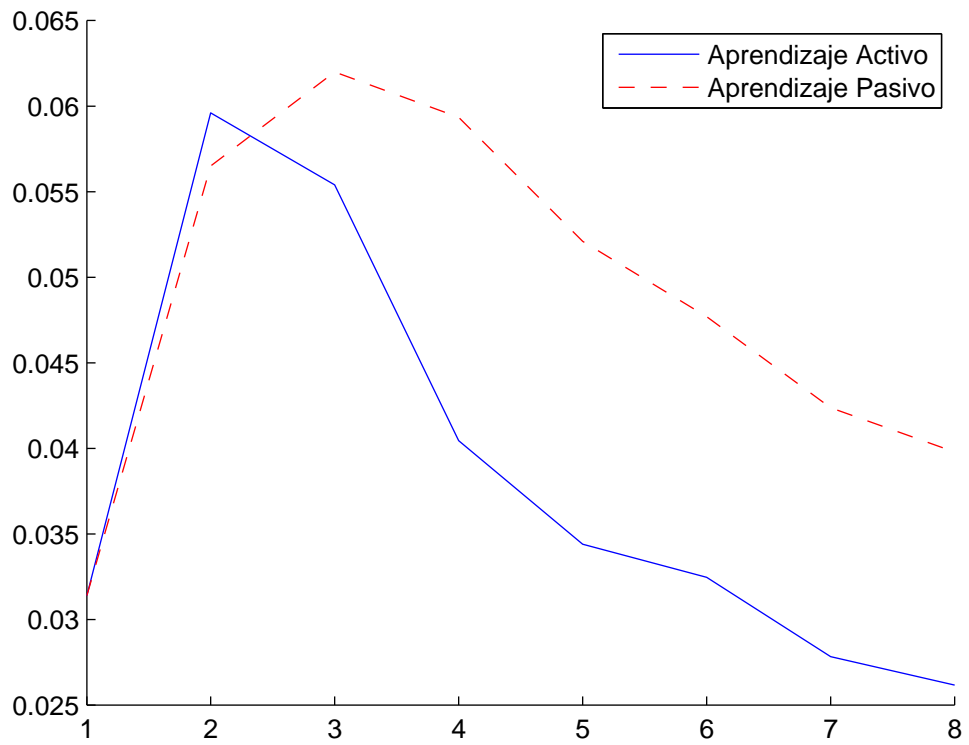


Figura 7.4: Error en modo activo(continua) y modo pasivo(punteada), concepto (Carrefour)

Capítulo 8

DISCUSIÓN

Durante este proyecto se realizó una investigación del estado del arte en procesamiento de imágenes y aprendizaje de conceptos en imágenes. Se planteó un requerimiento de generalización de descriptores y de tamaño de los mismos los cuales guiaron el trabajo en una dirección. Aunque se han planteado diferentes formas de abordar el problema más complejas, la metodología de desarrollo usada permitió cumplir los requerimientos de descripción de características de forma. El proyecto se puede considerar como un primer acercamiento al problema de llegar a una descripción de formas generalizada ya que estas están restringidas a curvas poligonales cerradas, el cual es un caso particular de las formas que se suelen encontrar en la vida real. De la misma manera el sistema de ranking desarrollado implementa algoritmos en los cuales se le da prioridad a la sencillez algorítmica, por lo que es posible que se pueda llegar a mejores resultados con un mejor método de ajuste de los parámetros del sistema de entrenamiento, aun así el sistema de aprendizaje activo basado en diversidad mostró algunas mejoras en las iteraciones necesarias para llegar a cierto nivel de error en algunos casos, ya que en otros casos no había mejoras significativas. No obstante diferentes métodos de aprendizaje activo funcionales al problema de imágenes en específico podrían ser abordados en un futuro para obtener resultados más robustos. El uso de Support Vector Machines y de Kernel Gaussiano generó en gran los resultados positivos debido al mapeo de los datos en un espacio de Hilbert. Finalmente, aunque el sistema estaba pensado para ser ejecutado en tiempo real por un usuario, el resultado obtenido es totalmente ineficiente en este sentido y requeriría del uso de grandes recursos computacionales para lograr este propósito, por lo que el sistema es aplicable en entornos de desarrollo mas no en entornos de aplicación comercial.

Capítulo 9

CONCLUSIONES

Este proyecto logró desarrollar satisfactoriamente un descriptor propuesto nuevo que cumple restricciones de un descriptor de alta complejidad tales como invarianza a rotación y tamaño. Por otra parte la eficiencia del descriptor desarrollado depende del ajuste del parámetro de versión de la curva según su número de vértices; aunque este parámetro debería ser variable para cada curva, esto causaría pérdida de unicidad en la representación de las curvas e imposibilidad de normalización de cada una de sus dimensiones, así queda por solucionar este problema de representación variable según la complejidad de la curva sin que haya pérdida de unicidad de la información. El producto final dista de uso práctico debido a su velocidad de procesamiento de información y a la restricción de conceptos que se pueden aprender, la cual es por ahora solo una aproximación investigativa y de desarrollo interno.

Aun así, el uso de Support Vector Machines con kernel Gaussiano y un método de aprendizaje activo entregó resultados de rendimiento aceptables para diferentes conceptos aprendidos durante el proyecto. Por lo que los objetivos del proyecto se cumplieron en su totalidad a partir de la metodología planteada y siguiendo un proceso de validación de resultados apropiado.

Bibliografía

- [1] Li-Wei Kang, Chao-Yung Hsu, Hung-Wei Chen, and Chun-Shien Lu. Secure sift-based sparse representation for image copy detection and recognition. In *Multimedia and Expo (ICME), 2010 IEEE International Conference on*, pages 1248–1253, july 2010.
- [2] Sumei Dai, Tao Jiang, and Ruiqing Jia. Image recognition method of liquid column manometer readings. In *Multimedia Technology (ICMT), 2010 International Conference on*, pages 1–5, oct. 2010.
- [3] Yang Hu, Mingjing Li, and Nenghai Yu. Multiple-instance ranking: Learning to rank images for image retrieval. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:1–8, 2008.
- [4] F. Jurie and C. Schmid. Scale-invariant shape features for recognition of object categories. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–90–II–96 Vol.2, june-2 july 2004.
- [5] Longin Jan Latecki and Rolf Lakaè Mper. Shape similarity measure based on correspondence of visual parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1185–1190, 2000.
- [6] Tat-Seng Chua, Wai-Chee Low, and Chun-Xin Chu. Relevance feedback techniques for color-based image retrieval. In *Multimedia Modeling, 1998. MMM '98. Proceedings. 1998*, pages 24–31, oct 1998.
- [7] B. Patrice and H. Konik. Texture similarity queries and relevance feedback for image retrieval. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 4, pages 55–58 vol.4, 2000.
- [8] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 130–136, jun 1997.

- [9] Y. Yuan and K. Barner. An active shape model based tactile hand shape recognition with support vector machines. In *Information Sciences and Systems, 2006 40th Annual Conference on*, pages 1611–1616, march 2006.
- [10] Aiwen Jiang, Chunheng Wang, and Yuanping Zhu. Calibrated rank-svm for multi-label image categorization. In *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 1450–1455, june 2008.
- [11] M. Sonka. *Image processing, analysis, and machine vision*. PWS Pub., 1999.
- [12] Longin Jan Latecki and Rolf Lakämper. Contour-based shape similarity. In Dionysius Huijsmans and Arnold Smeulders, editors, *Visual Information and Information Systems*, volume 1614 of *Lecture Notes in Computer Science*, pages 657–657. Springer Berlin / Heidelberg, 1999.
- [13] Longin Jan Latecki and Rolf Lakämper. Polygon evolution by vertex deletion. In *Proceedings of the Second International Conference on Scale-Space Theories in Computer Vision*, SCALE-SPACE '99, pages 398–409, London, UK, UK, 1999. Springer-Verlag.
- [14] Longin Jan Latecki and Rolf Lakämper. Convexity rule for shape decomposition based on discrete contour evolution. *Computer Vision and Image Understanding*, 73:441–454, 1999.
- [15] Corinna Cortes and Vladimir Vapnik. Support-vector networks. In *Machine Learning*, pages 273–297, 1995.
- [16] Yongmin Li, Shaogang Gong, and H. Liddell. Support vector regression and classification based multi-view face detection and recognition. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 300–305, 2000.
- [17] James S. Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2546–2554. 2011.
- [18] Olivier Chapelle and Vladimir Vapnik. Model selection for support vector machines, 2000.
- [19] S. Agarwal and D. Roth. Learnability of bipartite ranking functions. In *COLT*, pages 16–31, 2005.
- [20] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 133–142, New York, NY, USA, 2002. ACM.

- [21] Simon Tong and Edward Chang. Support vector machine active learning for image retrieval. In *Proceedings of the ninth ACM international conference on Multimedia*, MULTIMEDIA '01, pages 107–118, New York, NY, USA, 2001. ACM.
- [22] P.H. Gosselin and M. Cord. Active learning methods for interactive image retrieval. *Image Processing, IEEE Transactions on*, 17(7):1200–1211, july 2008.
- [23] S.C.H. Hoi, Rong Jin, Jianke Zhu, and M.R. Lyu. Semi-supervised svm batch mode active learning for image retrieval. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7, june 2008.
- [24] S.C.H. Hoi, Rong Jin, and M.R. Lyu. Batch mode active learning with applications to text categorization and image retrieval. *Knowledge and Data Engineering, IEEE Transactions on*, 21(9):1233–1248, sept. 2009.
- [25] Sanjoy Dasgupta. The two faces of active learning. In *Proceedings of the 12th International Conference on Discovery Science*, DS '09, pages 35–35, Berlin, Heidelberg, 2009. Springer-Verlag.
- [26] Hwanjo Yu. Svm selective sampling for ranking with application to data retrieval. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 354–363, New York, NY, USA, 2005.
- [27] S. Rajaram, C.K. Dagi, N. Petrovic, and T.S. Huang. Diverse active ranking for multimedia search. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, june 2007.
- [28] B.W. Silverman. *Density Estimation for Statistics and Data Analysis*. Monographs on Statistics and Applied Probability. Chapman and Hall, 1986.
- [29] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, June 1986.
- [30] James. Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305, 2012.



SISTEMA DE BIBLIOTECAS
IDENTIFICACIÓN TRABAJO DE GRADO

FECHA DE ELABORACIÓN		
DD	MM	AAAA
13	06	2012

NIT: 860.007.386-1

1. IDENTIFICACIÓN AUTOR(ES) DEL TRABAJO DE GRADO					
CÓDIGO	DOCUMENTO DE IDENTIDAD TIPO	NÚMERO	APELLIDOS	NOMBRES	CORREO ELECTRÓNICO
200812653	CC	1026274883	Mejía Mora	Hernán Darío	hd.mejia27@uniandes.ec
	CC				
	CC				
	CC				
	CC				
	CC				

PROGRAMA	Pregrado	
FACULTAD	Facultad de Ingeniería	
DEPARTAMENTO	Departamento de Ingeniería Eléctrica y Electrónica	

ENTREGÓ FORMATO:

SB-10 "Entrega trabajo de grado y autorización de uso a favor de la Universidad de los Andes". Documento con el cual, el autor permite que su trabajo sea utilizado por la Universidad, para fines de SB-10: consulta y de mención en sus catálogos bibliográficos, tanto físicos como en línea.

1.1 IDENTIFICACIÓN DE TRABAJO DE GRADO PARA DOBLE TITULACIÓN		
PROGRAMA	No Aplica	
FACULTAD	No Aplica	
DEPARTAMENTO	No Aplica	

TESIS PARA DOBLE TITULACIÓN:

Si el trabajo de grado presentado aplica para obtener dos (2) titulaciones, por favor marque esta casilla y diligencie la información de esta sección.

2. INFORMACIÓN GENERAL DEL TRABAJO DE GRADO		
TÍTULO DEL TRABAJO DE GRADO: Adquisición de características de forma y Ranking de Contenido en Video usando Support Vector Machines y Aprendizaje Activo		
DESCRIPCIÓN FÍSICA	MATERIAL ACOMPAÑANTE (Cantidad):	FECHA DE ELABORACIÓN
Número de páginas: 34	Caseteros Audio: Discos compactos:	DD MM AAAA
Ilustraciones: 12	Caseteros Video: Diapositivas:	30 05 2012
	Disquetes: Otros: ¿Cuáles?	
*RESUMEN DEL TRABAJO DE GRADO: El presente proyecto ahonda en la problemática de detección de patrones de forma en videos y desarrolla una metodología de solución al problema. Se presentan diferentes soluciones y se llega a una solución propia que satisface los objetivos de desempeño planteados. El proyecto igualmente se considera una implementación a nivel académico que busca validar diferentes herramientas de aprendizaje de máquinas y deja abierta la discusión para un desarrollo más apropiado para la implementación comercial.		
OBJETIVOS DEL TRABAJO DE GRADO: Implementar exitosamente un sistema computacional que realice un ordenamiento (ranking) de relevancia en contenido a bancos de videos basado en un requerimiento (query) específico para cada caso.		
METODOLOGÍA DEL TRABAJO DE GRADO: La metodología del proyecto consistió en una investigación bibliográfica sobre el estado del arte de los temas tratados, posteriormente el planteamiento de una metodología propia de abstracción de características y finalmente la implementación de los algoritmos del proyecto.		
CONCLUSIONES DEL TRABAJO DE GRADO: Fue posible desarrollar un descriptor de formas que cumpliera con todos los requerimientos y restricciones planteados aunque la eficiencia del mismo es dependiente de la complejidad escogida para su representación. Fue posible implementar un sistema que aprende satisfactoriamente diversos conceptos de forma presentes en imágenes para llegar a sistemas que aplican un ordenamiento por puntaje al contenido en videos a partir de su relevancia a un concepto determinado.		

*PALABRAS CLAVES (TEMAS) DEL TRABAJO DE GRADO:

Content based information retrieval, procesamiento de imágenes, Image Processing, ranking, Support Vector Machines, Active Learning, Aprendizaje Activo, information Retrieval, descriptores de forma, shape descriptors,

ACUERDOS DE CONFIDENCIALIDAD: NO TIENE ACUERDO(S) TIENE ACUERDO(S)

Si selecciona tener acuerdo de confidencialidad, por favor diligencie el siguiente cuadro:

Persona natural o jurídica	Desde DD MM AAAA	Hasta DD MM AAAA

3. FIRMAS

AUTORES (Nombre completo)	*FIRMAS
Hernán Darío Mejía Mora	Hernán D. Mejía M.
DIRECTORES / ASESORES (Nombre completo)	*FIRMAS
Fernando Lotano M.	F.L.
JURADO / LECTOR (Nombre completo)	*FIRMAS
Reinaldo Augusto Uribe Muriel	R.A.U.

Las firmas de Autor y Director/Asesor son obligatorias. Si tiene inconvenientes con el registro de la firma del Jurado/Lector, deberá tramitar ante la respectiva Facultad la autorización para registrar las firmas de pares o un sello que justifique la ausencia de la firma faltante.

SB-09

Verificar Información Imprimir



ENTREGA EJEMPLAR TRABAJO DE GRADO Y AUTORIZACIÓN DE SU USO A FAVOR DE LA UNIVERSIDAD DE LOS ANDES

Yo Hernán Darío Mejía Mora, mayor de edad, vecino de Bogotá D.C., identificado con la Cédula de Ciudadanía N° 1026274883 de Bogotá, actuando en nombre propio, en mi calidad de autor del trabajo de tesis, monografía o trabajo de grado denominado: ADQUISICIÓN DE CARACTERÍSTICAS DE FORMA Y RANKING DE CONTENIDO EN VIDEO USANDO SUPPORT VECTOR MACHINES Y APRENDIZAJE ACTIVO

Hernán Darío Mejía Mora, hago entrega del ejemplar respectivo y de sus anexos del ser el caso, en formato digital o electrónico (CD-ROM) y autorizo a LA UNIVERSIDAD DE LOS ANDES, para que en los términos establecidos en la Ley 23 de 1982, Ley 44 de 1993, Decisión Andina 351 de 1993, Decreto 460 de 1995 y demás normas generales sobre la materia, utilice y use en todas sus formas, los derechos patrimoniales de reproducción, comunicación pública, transformación y distribución (alquiler, préstamo público e importación) que me corresponden como creador de la obra objeto del presente documento. PARÁGRAFO: La presente autorización se hace extensiva no sólo a las facultades y derechos de uso sobre la obra en formato o soporte material, sino también para formato virtual, electrónico, digital, óptico, usos en red, internet, extranet, intranet, etc., y en general para cualquier formato conocido o por conocer.

EL AUTOR - ESTUDIANTES, manifiesta que la obra objeto de la presente autorización es original y la realizó sin violar o usurpar derechos de autor de terceros, por lo tanto la obra es de su exclusiva autoría y tiene la titularidad sobre la misma. PARÁGRAFO: En caso de presentarse cualquier reclamación o acción por parte de un tercero en cuanto a los derechos de autor sobre la obra en cuestión, EL ESTUDIANTE - AUTOR, asumirá toda la responsabilidad, y saldrá en defensa de los derechos aquí autorizados; para todos los efectos la Universidad actúa como un tercero de buena fe.

Para constancia se firma el presente documento en dos (02) ejemplares del mismo valor y tenor, en Bogotá D.C., a los Quince 15 días del mes de Junio de Dos Mil Doce 20 12.

EL AUTOR - ESTUDIANTE.

(Firma) Hernán D. Mejía M.

Nombre Hernán Darío Mejía Mora

C.C. N° 1026274883 de Bogotá

SB-10