

ELIMINACIÓN DE FOTOGRAMAS NO INFORMATIVOS EN UN SISTEMA DE ANOTACIÓN DE VIDEOS DE COLONOSCOPIA

CRISTIAN BALLESTEROS

DIRECTORA
MARIA TRUJILLO, PH.D.

CO-DIRECTORA
CLAUDIA MAZO, ING.



Escuela de Ingeniería de Sistemas y Computación
Facultad de Ingeniería
Universidad del Valle

Enero 2016

RESUMEN

Con los recientes avances tecnológicos se han logrado desarrollos significativos en diversos campos, como ha ocurrido en el campo de la salud. Estos desarrollos han permitido optimizar procesos que anteriormente se realizaban de forma manual o de los que se obtenían resultados subjetivos. En este trabajo de grado se abordan los temas de anotación de lesiones y clasificación de fotogramas no informativos en videos de colonoscopia, el cual está inspirado por el trabajo que se realiza en la Universidad de Dundee, Reino Unido, sobre el desarrollo de algoritmos de inteligencia artificial para la detección automática de cáncer y pólipos en videos de colonoscopia.

ABSTRACT

Recent technological advances have made significant developments in various fields, such as health. These developments have enabled optimising processes that were previously done manually or subjectively yielding subjective results. This final career work addresses the topics of annotating lesions such as cancer and polyps, and classifying non-informative frames in video of colonoscopy. The project is inspired by a work with the University of Dundee, UK, on developing machine learning algorithms for detecting automatically cancer and polyps on colonoscopy videos.

*Todos somos genios. Pero si juzgas a un pez
por su capacidad de escalar árboles,
vivirá toda su vida creyendo que es inútil.*

— Albert Einstein

AGRADECIMIENTOS

A mi madre *Amanda Ballesteros* por el apoyo incondicional y la motivación que me brinda en los momentos difíciles.

A la profesora *María Trujillo* por todas sus enseñanzas, consejos, gestión y ayuda en el desarrollo de este trabajo de grado.

Al profesor *Emmanuel Trucco*, al Dr. *Ruixuan (Roy) Wang* y a *Siyamalan Manivannan* por el aporte en el diseño de la aplicación y el protocolo para las anotaciones.

Al Ing. *Ivan Mejía* por toda la ayuda y soporte brindado en el desarrollo del sistema de anotaciones.

Al Dr. *Jesus A. Hoyos* por la grabación y anotación de los videos de colonoscopia.

Al Grupo de *Multimedia y Visión por Computador* y en especial a *Deisy Chaves* y *Claudia Mazo* por toda la ayuda y colaboración prestada.

ÍNDICE GENERAL

1	INTRODUCCIÓN	1
1.1	Enunciado del Problema	2
1.2	Justificación del Problema	2
1.2.1	Justificación Social	2
1.2.2	Justificación Académica	3
1.3	Objetivos	3
1.3.1	Objetivo General	3
1.3.2	Objetivos Específicos	3
1.4	Alcance	3
2	MARCO TEÓRICO	4
2.1	Conceptos de Procesamiento Digital de Imágenes	4
2.1.1	Espacio de Color	4
2.1.2	Histograma de una Imagen	5
2.1.3	Umbralización	5
2.1.4	Filtros de Paso Alto y Bajo	6
2.1.5	Bordes de una Imagen	6
2.1.6	Operaciones Morfológicas	9
2.1.7	Transformada Discreta de Fourier	9
2.1.8	Textura de una Imagen	10
2.1.9	k-means	12
2.1.10	Support Vector Machine (SVM)	13
2.2	Tecnologías para el Desarrollo Web Multimedia	14
2.2.1	HTML5	14
2.2.2	Javascript	15
2.2.3	JQuery	15
2.2.4	Boopstrap	15
2.2.5	AJAX	16
2.2.6	JW Player	16
2.2.7	FFMPEG	16
2.2.8	RTMP	17
2.2.9	PHP	17
2.2.10	NGINX Server	17
2.2.11	PostgreSQL	18
2.2.12	eXtreme Programming (eXtreme Programming (XP))	18
3	ESTADO DEL ARTE	19
3.1	Clasificación de Fotogramas en Videos de Colonosco- pia	19
3.1.1	Clasificación Basada en Detección de Bordes	19
3.1.2	Clasificación Basada en Patrones de Textura	22
3.1.3	Otros Métodos de Clasificación	23
3.2	Herramientas para Anotación de Videos	24

3.2.1	Open Video Annotation Project (OVA)	24
3.2.2	Advene Project	25
3.2.3	Anotating Academic Video (AAV)	26
3.2.4	Video Image Annotation Tool (VIA)	27
4	CLASIFICACIÓN AUTOMÁTICA DE FOTOGRAMAS EN VIDEOS DE COLONOSCOPIA	28
4.1	Clasificación Basada en Detección de Bordes	29
4.1.1	Clasificación por Detección de Bordes	30
4.1.2	Segmentación de Brillo	32
4.2	Clasificación Basada en Descriptores de Textura	33
4.2.1	Preprocesamiento	34
4.2.2	Análisis de Textura	35
4.2.3	Clasificación de Fotogramas con k-means	36
4.2.4	Clasificación de Fotogramas con SVM	37
4.3	Evaluación de las Técnicas	37
4.3.1	Conjunto de Datos	37
4.3.2	Medidas de Desempeño	38
4.3.3	Evaluación del Método de Detección de Bordes	39
4.3.4	Evaluación del Método de Análisis de Textura	40
4.3.5	Análisis y Comparación de los Métodos	44
4.3.6	Comparación entre Clasificación Supervisada y No Supervisada	45
4.3.7	Elección del Mejor Método de Clasificación	45
5	HERRAMIENTA DE ANOTACIÓN DE VIDEOS DE COLONOSCOPIA	46
5.1	Propuesta General de la Aplicación	46
5.2	Selección de Herramientas de Desarrollo	46
5.3	Implementación	47
5.3.1	Actores	48
5.3.2	Arquitectura	48
5.3.3	Módulos	48
5.3.4	Historias de Usuario	50
5.3.5	Plan de Entregas - "Release Plan"	51
5.3.6	Interfaz Gráfica	52
5.4	Integración de la Aplicación con los Métodos de Clasificación de Fotogramas	57
5.5	Pruebas de Aceptación	57
6	CONCLUSIONES	60
6.1	Conclusiones sobre los Métodos de Clasificación	60
6.2	Conclusiones sobre la Aplicación de Anotaciones	62
7	TRABAJO FUTURO	63
	BIBLIOGRAFÍA	65

ÍNDICE DE FIGURAS

Figura 1	Ejemplo de Umbralización de un fotograma de colonoscopia utilizando el algoritmo Otsu. (a) Imagen original en escala de grises. (b) Imagen segmentada con el algoritmo Otsu. 5	
Figura 2	Bordes de un fotograma de colonoscopia. (a) Imagen original en escala de grises. (b) Bordes obtenidos con Sobel. (c) Bordes obtenidos con Canny 6	
Figura 3	Matrices de convolución de Sobel 7	
Figura 4	Direcciones consideradas para los bordes en Canny. Tomado de [11]. 8	
Figura 5	Tipos de elementos estructurales. Tomado de [33]. 9	
Figura 6	Ejemplo de dilatación de una imagen binaria. Tomado de [33]. 10	
Figura 7	Cálculo de la matriz de co-ocurrencia. Tomado de [29]. 11	
Figura 8	Algoritmo del operador LBP 12	
Figura 9	Ejemplo de particionamiento con el algoritmo k-means. Los puntos de color negro corresponden a los centroides de cada grupo. 13	
Figura 10	Hiperplanos lineales en un espacio bidimensional para un conjunto de datos separables en dos grupos. (a) Ejemplos de hiperplanos. (b) Hiperplano óptimo. Tomado de [9]. 14	
Figura 11	“Hola Mundo” en HTML5 utilizando: (a) Javascript, (b) jQuery. pruebas ejecutadas en jsfiddle.net 16	
Figura 12	(a) Fotograma No Informativo, (b) y (c) bordes de (a), (d) detalles de los bordes de ruido, (e) Fotograma Informativo, (f) y (g) bordes de (e), y (h) detalle de los bordes limpios o bien definidos. Tomado de [34]. 19	
Figura 13	Fotograma Ambiguo (a) y sus bordes (b). Tomado de [34]. 20	
Figura 14	(a) Fotograma no informativo con zonas de brillo, (b) Área de brillo absoluta de (a), (c) Área relativa de brillo de (a), (d) Mapa de reflexión de brillo de (a), y (e) mejoramiento de bordes de (a) eliminando los bordes de las zonas de brillo. Tomado de [34]. 22	

Figura 15	Proceso general de la técnica de clasificación por patrones de textura propuesta en [34]	22
Figura 16	(a) Fotograma Informativo y su representación en el dominio de Fourier. (b) Fotograma No Informativo y su representación en el dominio de Fourier. <i>Tomado de [34].</i>	23
Figura 17	Primera fila: Fotogramas originales obtenidos con una cápsula inalámbrica. Segunda fila: Fotogramas segmentados con <i>Mean Shift</i> . <i>Tomado de [49].</i>	24
Figura 18	Interfaz gráfica de OVA	25
Figura 19	Interfaz gráfica de Advene	25
Figura 20	Interfaz gráfica de Anotating Academic Video	26
Figura 21	Interfaz gráfica de VIA	26
Figura 22	Ejemplos de fotogramas en videos de colonoscopia: (a) Fotograma Informativo. (b) Fotograma No Informativo debido al contacto de la cámara con las paredes del colon. (c) Fotograma No Informativo por borrosidad. (d) Fotograma No Informativo por reflexión de la luz con el agua.	28
Figura 23	Bordes en fotogramas de videos de colonoscopia: (a) Fotograma Informativo. (b) Bordes de (a). (c) Fotograma No Informativo debido al contacto de la cámara con las paredes del colon. (d) Bordes de (c). (e) Fotograma No Informativo por borrosidad. (f) Bordes de (e). (g) Fotograma No Informativo por reflexión de la luz con el agua. (h) Bordes de (g).	29
Figura 24	Diagrama general del método propuesto basado en detección de bordes	30
Figura 25	Proceso de la clasificación por detección de bordes	31
Figura 26	Técnica de segmentación de brillo	32
Figura 27	Análisis de brillo en los fotogramas. (a) Fotograma Informativo con pocas zonas de brillo. (b) Histograma de (a). (c) Fotograma no informativo con altas zonas de brillo. (d) Histograma de (c). El rango entre 205 y 255 se muestra en color blanco con fondo azul, resaltando las frecuencias donde se concentra el brillo en las imágenes	33
Figura 28	(a) Fotograma No Informativo. (b) Espectro de Frecuencias de (a). (c) Fotograma Informativo. (d) Espectro de Frecuencias de (c)	35

Figura 29	Diagrama general de la clasificación de los fotogramas de colonoscopia con <i>k-means</i>	36
Figura 30	Interfaz gráfica de la aplicación web para la selección de los fotogramas.	38
Figura 31	Condiciones de evaluación para la clasificación por análisis de textura	41
Figura 32	Rendimiento de los métodos de clasificación	44
Figura 33	Integración de las tecnologías seleccionadas	47
Figura 34	Primer diseño en <i>Mocqup</i> de la interfaz gráfica de la aplicación de anotaciones CAT	48
Figura 35	Arquitectura general de CAT	49
Figura 36	Ventana de ingreso a la aplicación	52
Figura 37	Ventana general de la aplicación para anotaciones	53
Figura 38	Ventana de la aplicación cuando se ha seleccionado un video	53
Figura 39	Detalle de la barra de control de reproducción	54
Figura 40	Selección de un intervalo de anotación	54
Figura 41	Vista de la aplicación en el evento de editar una anotación	54
Figura 42	Ventana principal del administrador	55
Figura 43	Cuadro de diálogo para el evento de agregar un nuevo usuario	55
Figura 44	Cuadro de diálogo para el evento de agregar un nuevo video	56
Figura 45	Ventana de la aplicación para el evento de asignar un video a un especialista	56
Figura 46	Detalle de la asignación de un video a un especialista	56
Figura 47	Enlace <i>annotations</i> para descargar el reporte de anotación	56

ÍNDICE DE TABLAS

Tabla 1	Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en detección de bordes con y sin la técnica de segmentación de brillo y el método de Canny propuesto en [34]	40
Tabla 2	Medidas de rendimiento del método propuesto basado en detección de bordes con y sin la técnica de segmentación de brillo y el método de Canny propuesto en [34], utilizando los datos de la Tabla 1	40
Tabla 3	Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes.	41
Tabla 4	Medidas de rendimiento del método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes., utilizando los datos de la Tabla 3	42
Tabla 5	(Predicción) Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes.	42
Tabla 6	(Generalización) Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes.	43
Tabla 7	(Predicción) Medidas de rendimiento del método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes., utilizando los datos de la Tabla 5	43
Tabla 8	(Generalización) Medidas de rendimiento del método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes., utilizando los datos de la Tabla 6	43
Tabla 9	Listado de historias de usuario	50
Tabla 10	Plan de entregas para el desarrollo de la aplicación CAT	51
Tabla 11	Pruebas de aceptación de la aplicación CAT	58

LISTA DE ALGORITMOS

1	Clasificación de fotogramas por detección de bordes . .	31
2	Cálculo del porcentaje de píxeles de borde en una imagen	31
3	Técnica de segmentación de brillo	34

ABREVIATURAS

RGB Red Green Blue

HSV Hue Saturation Value

HSI Hue Saturation Intensity

EPP Edge Pixel Percentage

PBP Pixel Brightness Percentage

Th Threshold

LBP Local Binary Pattern

DFT Transformada Discreta de Fourier

SVM Support Vector Machine

CAT Colonoscopy Annotation Tool

XP eXtreme Programming

CRUD Create Read Update Delete

INTRODUCCIÓN

Cada año son diagnosticados más de un millón de nuevos casos de cáncer de colon en el mundo y se estima que para el 2035 la cifra sea de 2.4 millones [21]. El cáncer de colon, en el Reino Unido es el tercer cáncer más común [46] y en Colombia es el sexto más frecuente [14]. A pesar de que se desconocen las causas exactas que producen el cáncer colorrectal, existen factores que se asocian con dicha enfermedad como son la edad, antecedentes familiares, sobrepeso, falta de ejercicio y alcohol, entre otros [14]. Con la detección temprana del cáncer, el paciente tiene mayor posibilidad de sobrevivir, siendo más probable el éxito de los tratamientos para combatirlo. Uno de los mecanismos de prevención es la colonoscopia, la cual consiste en un examen exploratorio a través del intestino grueso con un instrumento conocido como colonoscopio, esta prueba es realizada por un médico gastroenterólogo quien luego se encarga de realizar el diagnóstico.

En este documento se presenta el desarrollo de una aplicación que facilita a los médicos realizar anotaciones y descartar los fotogramas no informativos en videos de colonoscopia, con lo que se pretende colaborar en la investigación realizada por el estudiante de Doctorado de la Universidad de Dundee, Siyamalan Manivannan, quien propone el reconocimiento automático de lesiones en videos de colonoscopia con técnicas de inteligencia artificial utilizando como base de conocimiento los fotogramas obtenidos de las anotaciones realizadas por los médicos [27],[28]. Los videos de las colonoscopias actualmente utilizados por Siyamalan en su investigación, que también son objeto de estudio en este trabajo, son tomados por el Dr. Jose Arbey Hoyos del Hospital Universitario del Valle, quien además brinda su colaboración en el desarrollo de la aplicación de anotaciones.

Este proyecto cuenta con los siguientes usuarios para apoyar el proceso de desarrollo de la aplicación: Dr. Jose Arbey Hoyos, del HUV, Profesor Emanuele Trucco, Universidad de Dundee, y Siyamalan Manivannan estudiante de doctorado de la Universidad de Dundee. Adicionalmente, el Ing. Heber Mejia de la Universidad Señor de Sipán, en Perú, brinda asesoría en procesos de desarrollo de software y el Ing. Sebastian Scotti de OpenHealth SAS, es un consultor especializado que apoya el proceso de implementación.

A continuación se presentarán la definición del problema, objetivos y alcance para la implementación de una herramienta que permite realizar anotaciones en videos de colonoscopia y que tiene la funcionalidad de descartar los fotogramas no informativos automáticamente.

1.1 ENUNCIADO DEL PROBLEMA

La colonoscopia es un procedimiento médico que permite al especialista examinar el tejido del colon. Este se lleva a cabo utilizando un colonoscopio, que consiste de una manguera flexible con fibras ópticas que transportan la luz al interior del colon y un *microchip* de televisión que recoge estas imágenes digitalizadas para observarlas en un monitor, imprimirlas en fotografías y grabarlas en video [6]. Al observar el video es posible determinar la presencia de cáncer o diferentes lesiones, como pólipos. Un médico gastroenterólogo se encarga de etiquetar un fragmento de video de acuerdo al tipo de lesión encontrada.

Herramientas de apoyo al diagnóstico automatizan el reconocimiento de lesiones, tales como cáncer y pólipos, utilizando técnicas de procesamiento digital de imágenes y algoritmos de aprendizaje supervisado con las anotaciones de los médicos como base de conocimiento.

Los algoritmos de inteligencia artificial construyen un modelo con base en un conjunto de entrenamiento, que es formado a partir de las anotaciones que los gastroenterólogos realicen sobre videos de colonoscopia. En la actualidad no existe ninguna herramienta que facilite el proceso de anotación sobre estos videos, lo que dificulta la construcción del conjunto base de conocimiento y el entrenamiento de los algoritmos, ya que su capacidad de generalización está determinada por dicho conjunto.

En los videos de colonoscopia se encuentran fotogramas que no aportan información al diagnóstico debido a que presentan borrosidad, bajo contraste, ruido o brillo, por lo tanto, si una anotación contiene este tipo de fotogramas puede causar alteración en el aprendizaje del algoritmo y consecuentemente, generar modelos que no clasifican de forma correcta las lesiones, ya que las técnicas de inteligencia artificial dependen fuertemente del conjunto de datos de entrenamiento.

En relación a lo anterior, en este trabajo de grado se aborda la clasificación de fotogramas de videos de colonoscopia, usando métodos de preprocesamiento digital de imágenes, para identificar aquellos que no contienen información que aporte diagnóstico y posteriormente eliminarlos. De esta manera se espera que las anotaciones tengan el contenido necesario para el entrenamiento de los algoritmos de inteligencia artificial.

1.2 JUSTIFICACIÓN DEL PROBLEMA

1.2.1 *Justificación Social*

El desarrollo de métodos de diagnóstico automático de cáncer de colon, basados en técnicas computacionales, tienen una motivación

social en la medida en que intentan disminuir los errores en el diagnóstico, incrementar los índices de detección temprana y aumentar los niveles de sobrevivencia de los pacientes afectados por esta enfermedad dada la elevada incidencia de este mal en la población mundial. En relación a lo anterior este proyecto toma esa misma motivación social, en la medida en que se intenta aportar de manera significativa a la eficiencia y correctitud de las anotaciones.

1.2.2 *Justificación Académica*

El desarrollo de este proyecto requiere de la aplicación de conceptos y técnicas aprendidas en el transcurso de la carrera: programación orientada a objetos, programación interactiva, aplicaciones en la web y redes inalámbricas, procesamiento de imágenes, desarrollo de software, inteligencia artificial, entre otras. Además, será necesario profundizar en nuevos temas, no solo en el campo del software y la ingeniería, sino también en el campo de la salud y del trabajo interdisciplinario.

1.3 OBJETIVOS

1.3.1 *Objetivo General*

Desarrollar una aplicación de anotación de videos de colonoscopias que tenga la funcionalidad de descartar automáticamente los fotogramas no informativos.

1.3.2 *Objetivos Específicos*

- Implementar una aplicación de anotaciones de video de colonoscopia.
- Proponer métodos para el reconocimiento automático de fotogramas no informativos en videos de colonoscopia.
- Descartar los fotogramas no informativos de los videos de colonoscopia.

1.4 ALCANCE

El prototipo de software será implementado usando *frameworks* que tengan funcionalidades para el desarrollo de aplicaciones multimedia. Los métodos de identificación y clasificación de fotogramas no informativos se desarrollarán teniendo en cuenta características como borrosidad, bajo contraste, ruido, brillo y contenido de color.

En este trabajo de grado se desarrolló una aplicación que facilita la anotación de lesiones en videos de colonoscopia y tiene la funcionalidad de descartar automáticamente los fotogramas no informativos de estos videos, para ello se requieren herramientas y conceptos que se describen en la siguiente revisión teórica.

2.1 CONCEPTOS DE PROCESAMIENTO DIGITAL DE IMÁGENES

2.1.1 *Espacio de Color*

Los espacios de color, también conocidos como modelos o sistemas de color, son modelos matemáticos que describen el rango de colores como tuplas de números[4]. Las imágenes digitales son almacenadas, manipuladas y transmitidas en una gran variedad de espacios de color, a continuación se describen aquellos que fueron empleados en este trabajo de grado.

2.1.1.1 *Escala de Grises*

Este espacio de color representa una imagen en un determinado rango de niveles de gris que van de desde el negro al blanco[12]. La escala de grises es el espacio de color más común para el tratamiento de imágenes con operaciones morfológicas y diferentes tipos de filtros, dado que el contenido de la imagen está representado en un solo canal y es más fácil emplear técnicas para el reconocimiento de objetos.

2.1.1.2 *Red Green Blue (RGB)*

RGB es la representación más común de color en una imagen. Este espacio es un modelo aditivo, donde cada nuevo color es construido aumentando (o disminuyendo) niveles a los 3 colores básicos: Rojo (*Red*), Verde (*Green*) y Azul (*Blue*)[32]. Por ejemplo, el amarillo es construido a partir de la mezcla entre rojo y verde, la pureza del nuevo color depende de las intensidades de cada color básico.

2.1.1.3 *Hue Saturation Value (HSV) y Hue Saturation Intensity (HSI)*

HSV y HSI son una representación geométrica del espacio de color RGB. Fue creado bajo la premisa de que es más intuitivo pensar en un color en términos de tono y saturación que en adición de niveles[32]. La diferencia de estos dos espacios está en que el primero tiene una

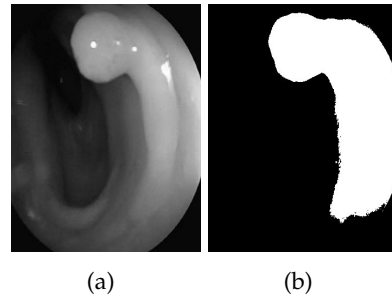


Figura 1: Ejemplo de Umbralización de un fotograma de colonoscopia utilizando el algoritmo Otsu. (a) Imagen original en escala de grises. (b) Imagen segmentada con el algoritmo Otsu.

componente para medir el brillo (V) y el otro mide la intensidad (I) del color. Estos modelos son ideales para algoritmos de procesamiento de imágenes basados en descriptores de color.

2.1.2 *Histograma de una Imagen*

El histograma representa de manera gráfica el comportamiento de los píxeles en cuanto a intensidad de color. Puede ser visto como un gráfico de barras donde el eje horizontal representa el total de intensidades de la imagen dependiendo de la escala de bits que se use, y el eje vertical corresponde a la frecuencia acumulada por cada valor de intensidad, es decir, al total de píxeles que tienen ese valor dentro de la imagen[47]. El histograma es utilizado para obtener un panorama general de la imagen y a simple vista conocer información sobre la distribución de color o la presencia de objetos.

2.1.3 *Umbralización*

La umbralización o segmentación consiste en cambiar la presentación de una imagen de tal forma que se simplifique su análisis y así lograr identificar objetos o límites. El cambio se realiza mediante la comparación de cada píxel de la imagen con un determinado umbral. Dependiendo de la condición, si el valor del píxel es mayor que el umbral se modifica por un valor (normalmente el más alto) en la escala que se esté manejando, en caso contrario se cambia por el valor opuesto en la escala[47]. El resultado de la umbralización es una imagen binaria. Esta técnica complementa la transformación global en la imagen y facilita la identificación de patrones.

La [Figura 1](#) muestra un ejemplo de umbralización de una imagen de colonoscopia utilizando un umbral generado por el algoritmo Otsu[37].

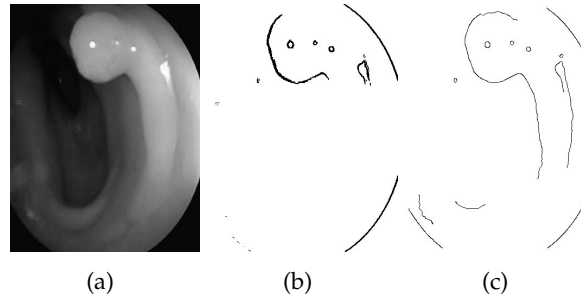


Figura 2: Bordes de un fotograma de colonoscopia. (a) Imagen original en escala de grises. (b) Bordes obtenidos con Sobel. (c) Bordes obtenidos con Canny

2.1.4 Filtros de Paso Alto y Bajo

Este tipo de filtros en procesamiento digital de imágenes son utilizados con el fin de mejorar la visualización de objetos y así extraer características de la imagen. En particular, los filtros de paso alto atenúan los píxeles de bajas frecuencias y mantienen los píxeles de altas frecuencias, incluso los pueden magnificar. Lo contrario ocurre con los filtros de paso bajo, quienes conservan las bajas frecuencias y eliminan las altas[30]. Normalmente el paso bajo es utilizado para suavizar la imagen o eliminar ruido, mientras que el paso alto intensifica los detalles.

Dentro de cada tipo de filtro se pueden encontrar algoritmos como:

- **Paso alto:** Detección de bordes y contorno tales como Sobel, Prewitt y Canny.
- **Paso bajo:** Suavizado y eliminación de ruido como el filtro promedio, media, mediana y Gaussiano.

2.1.5 Bordes de una Imagen

Los bordes de una imagen pueden ser definidos como los cambios drásticos de brillo entre píxeles, estos cambios también son conocidos como discontinuidades[30]. En análisis y procesamiento de imágenes la detección de bordes es una herramienta de gran importancia que se utiliza comúnmente para la detección y extracción de características. Existen diversos algoritmos para detectar bordes; a continuación se presentan los que fueron utilizados en este trabajo de grado.

2.1.5.1 Sobel

Sobel utiliza dos matrices de convolución de tamaño 3×3 para calcular la magnitud del gradiente en dirección horizontal y en dirección vertical (Figura 3). Al final los bordes son calculados con la mag-

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1
Horizontal			Vertical		

Figura 3: Matrices de convolución de Sobel

nitud del gradiente en ambas direcciones, utilizando la [Ecuación 1](#) y realizando una umbralización automática (generalmente aplicando el algoritmo Otsu)[2].

$$|G| = \sqrt{(G_x)^2 + (G_y)^2} \quad (1)$$

La [Figura 2b](#) ilustra los bordes obtenidos con Sobel de un fotograma de colonoscopia.

2.1.5.2 Canny

A continuación se presenta la revisión teórica del algoritmo Canny, la cual está basada en la descripción presentada en [2].

El algoritmo de detección de bordes Canny está basado en tres principios fundamentales para los detectores de bordes:

- Principio de Detección: Detectar, en lo posible, todos los bordes que existan en la imagen.
- Principio de Localización: La localización del borde según el algoritmo debe ser lo más aproximada a la localización real del borde en la imagen.
- Principio de Unicidad: Un borde debe marcarse solo una vez y, en lo posible, el ruido de la imagen no debe crear falsos bordes.

Canny se compone de tres grandes pasos: cálculo del gradiente, supresión no máxima e histéresis. A continuación se describe cada uno de ellos.

1. Cálculo del gradiente:

En este paso se calcula la magnitud ($|G|$) y dirección ϕ del gradiente. Al igual que en Sobel, se utilizan las matrices de convolución mostradas en la [Figura 3](#). La magnitud es calculada con la [Ecuación 1](#) y para el cálculo de la dirección es utilizada la [Ecuación 2](#). Previo a este primer paso, según las condiciones de la imagen, es recomendable utilizar alguna técnica de eliminación de ruido, como el filtro Gaussiano.

$$\phi = \arctan \frac{G_x}{G_y} \quad (2)$$

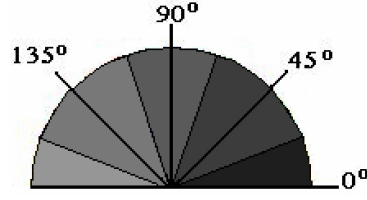


Figura 4: Direcciones consideradas para los bordes en Cany. Tomado de [11].

2. Supresión No Máxima:

En este paso se utilizan las dos imágenes obtenidas en el anterior, magnitud $|G|$ y dirección ϕ del gradiente. Inicialmente se consideran cuatro direcciones para ajustar los píxeles de $|G|$ a alguno de ellos. Las direcciones que se toman son (ver Figura 4): Este y Oeste 0° , Noreste y Suroeste 45° , Norte y Sur 90° y Noroeste y Sureste 135° . Se tiene que en la magnitud del gradiente los bordes son de valor alto, por lo tanto se forma una nueva imagen I con los píxeles de $|G|$ cuyo valor de magnitud es máximo. Los pasos para la supresión no máxima son los siguientes:

- **Cálculo de Direcciones:** Se obtiene una imagen de direcciones D de acuerdo a la Ecuación 3, donde se aproxima cada valor de $|G|$ a una de las direcciones consideradas.

$$G(x, y) = G(x, y) + 90 \quad (3)$$

- **Supresión de No Máximos:** Cada píxel $D(x, y)$ se compara con sus vecinos, si este es menor que al menos uno de ellos se le asigna el valor de 0, en caso contrario se le asigna el valor que tenga la magnitud del gradiente en esa posición $D(x, y) = G(x, y)$.

3. Histéresis:

Este paso final busca eliminar bordes creados a partir de ruido en la imagen y dejar solo los verdaderos bordes. Para llevar a cabo la histéresis se toma la imagen resultante del paso anterior S y dos umbrales, uno alto (T_{high}) y otro bajo (T_{low}). Se evalúa cada píxel de S en un mismo orden y para cada uno se localiza el siguiente píxel tal que $S(x, y) > T_{high}$ y, siguiendo la dirección del borde, se marca en la imagen de salida siempre que $S(x, y) > T_{low}$. Cada píxel evaluado se debe etiquetar como visto. El resultado de este paso son los bordes de la imagen, se debe tener especial cuidado con los umbrales seleccionados ya que se pueden eliminar verdaderos bordes.

La Figura 2(c) muestra los bordes de un fotograma de colonoscopia obtenidos con el algoritmo Cany, utilizando los umbrales $T_{high} = 5$ y $T_{low} = 1$.

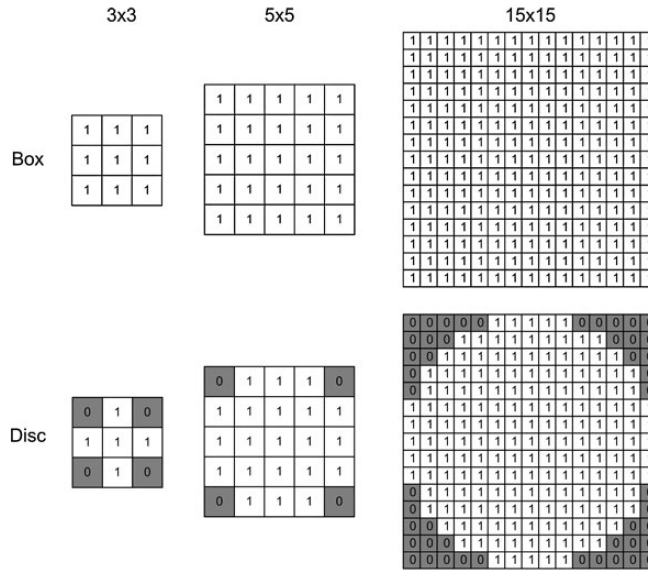


Figura 5: Tipos de elementos estructurales. Tomado de [33].

2.1.6 Operaciones Morfológicas

Las operaciones morfológicas son usadas comunmente en el tratamiento de imágenes digitales para identificar objetos a partir de la reconstrucción de imágenes distorcionadas o ruidosas[26]. En una operación morfológica cada pixel de la imagen es modificado de acuerdo a un elemento estructural que se define de manera previa y puede ser de cualquier forma geométrica.

La Figura 5 presenta algunos tipos de elementos estructurales que pueden ser usados en las operaciones morfológicas.

Existen dos operaciones morfológicas básicas: erosión y dilatación. En este trabajo de grado se utiliza la operación de dilatación que se describe a continuación.

2.1.6.1 Dilatación

El propósito de esta operación es aumentar el tamaño de los objetos en una imagen binaria. Al aplicar la dilatación se espera que los objetos pequeños se vuelvan grandes e incluso que se junten entre sí [33]. La Figura 6 ilustra un ejemplo de dilatación de una imagen binaria utilizando un elemento estructural rectangular de tamaño 3×3 .

2.1.7 Transformada Discreta de Fourier

La transformada discreta de Fourier es usada para descomponer una imagen en sus componentes de seno y coseno. El resultado de aplicar esta transformada es la representación de la imagen en el dominio de las frecuencias, también conocido como el dominio de

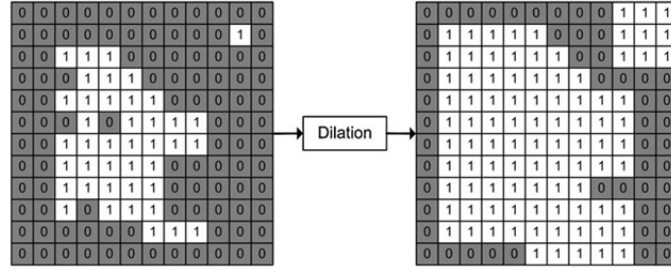


Figura 6: Ejemplo de dilatación de una imagen binaria. Tomado de [33].

Fourier. Cada punto del dominio de Fourier representa una frecuencia particular en el dominio espacial de la imagen [2].

La transformada de Fourier para una imagen de tamaño $M \times N$ es calculada de acuerdo a la Ecuación 4.

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp[-j2\pi(\frac{ux}{M} + \frac{vy}{N})] \quad (4)$$

La magnitud del espectro de Fourier es obtenido con el valor absoluto de la transformada $|F|$.

2.1.8 Textura de una Imagen

La textura es definida como un pequeño patrón elemental que se repite periódicamente en una imagen [22]. El ojo humano es capaz de reconocer una textura fácilmente, sin embargo para el computador es un proceso complejo. Con el fin de identificar patrones y características en las texturas se han desarrollado diferentes técnicas, algunas de ellas son mencionadas y utilizadas en este trabajo de grado y son definidas a continuación:

2.1.8.1 Matriz de Co-Ocurrencia

La matriz de co-ocurrencia describe la frecuencia con la que un valor en nivel de gris aparece en una relación espacial específica con otro valor de gris. Para el cálculo de la matriz de co-ocurrencia es necesario especificar las direcciones y distancia para el conteo. Generalmente se calcula la co-ocurrencia a la distancia de 1 pixel en cualquiera de las direcciones: Este y Oeste 0° , Noreste y Suroeste 45° , Norte y Sur 90° y Noroeste y Sureste 135° [29]. La Figura 7 muestra el cálculo de la matriz de co-ocurrencia GLCM, con una distancia de 1 en dirección Este, para una imagen I de tamaño 4×5 . La posición $(1, 1)$ en GLCM tiene el valor de 1 porque en la imagen original solo aparece una vez el 1 a la derecha de otro 1. Un razonamiento similar se realiza con la posición $(1, 2)$.

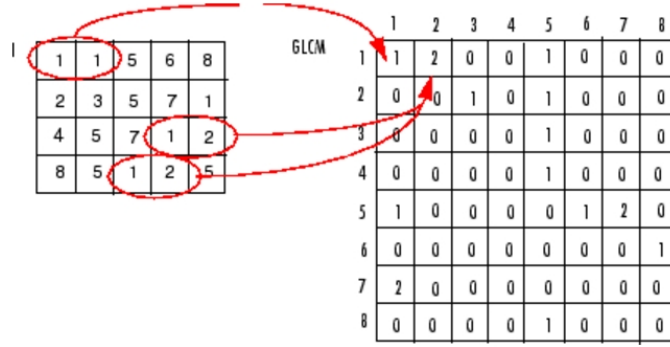


Figura 7: Cálculo de la matriz de co-ocurrencia. Tomado de [29].

Con el fin de obtener características de textura, se definen 14 medidas sobre la matriz de co-ocurrencia, llamadas las Métricas de Haralick [19]. De estas 14 métricas son 7 las más empleadas, las cuales se definen a continuación.

- Angular Second Moment (ASM)

$$\sum_i \sum_j \text{GLCM}(i, j)^2 \quad (5)$$

- Contraste

$$\sum_i \sum_j (i, j)^2 \times \text{GLCM}(i, j) \quad (6)$$

- Correlación

$$\sum_i \sum_j \frac{(i - \mu_x)(j - \mu_y) \text{GLCM}(i, j)}{\sigma_x \sigma_y} \quad (7)$$

- Disimilaridad

$$\sum_i \sum_j \text{GLCM}(i, j) \times |i - j| \quad (8)$$

- Entropía

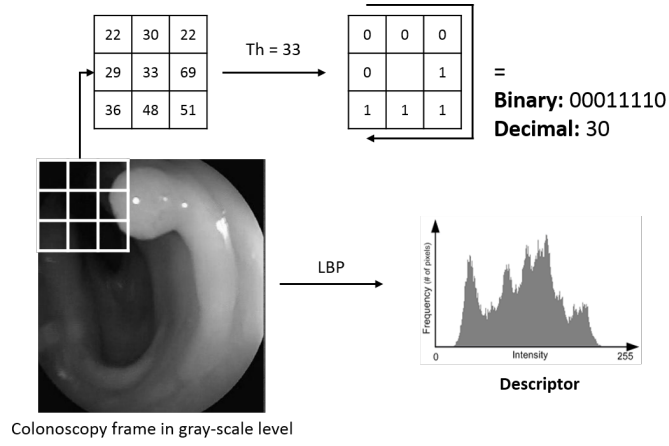
$$\sum_i \sum_j \text{GLCM}(i, j) \log \text{GLCM}(i, j) \quad (9)$$

- Energía

$$\sqrt{\text{ASM}} \quad (10)$$

- Uniformidad

$$\sum_i \sum_j \frac{\text{GLCM}(i, j)}{1 - |i - j|} \quad (11)$$

Figura 8: Algoritmo del operador **LBP**

donde:

$$\begin{aligned}\mu_x &= \sum_i \sum_j i \times \text{GLCM}(i, j), \quad \mu_y = \sum_i \sum_j j \times \text{GLCM}(i, j), \\ \sigma_x &= \sqrt{\sum_i \sum_j (i - \mu_x)^2 \times \text{GLCM}(i, j)} \quad y \\ \sigma_y &= \sqrt{\sum_i \sum_j (j - \mu_y)^2 \times \text{GLCM}(i, j)}.\end{aligned}$$

2.1.8.2 Local Binary Pattern (**LBP**)

LBP es un descriptor de textura muy popular, utilizado para clasificación. Consiste en modificar la imagen original con una ventana de convolución a través de umbralizaciones entre el pixel de interés y sus vecinos. Originalmente el operador **LBP** trabaja con un bloque de tamaño 3x3, los pixeles encerrados por el bloque son umbralizados con el valor de pixel central. El resultado de la umbralización es una cadena binaria que se convierte en un número decimal y se reemplaza con el pixel de interés. Finalmente se calcula un histograma de la nueva imagen el cual es tomado como descriptor.

La [Figura 8](#) describe el algoritmo del operador **LBP**.

2.1.9 *k-means*

K-means es un método de clasificación que particiona un grupo de datos en k sub-grupos, donde k es definido por el usuario al inicio del algoritmo. K-means busca obtener, del conjunto de datos, aquellos que son más representativos para cada k grupo, a estos datos se les conoce como centroides [2]. El objetivo es reunir en un mismo grupo todos los elementos que tengan la menor distancia entre ellos y un centroide. Generalmente se utiliza la distancia Euclidiana.

El algoritmo se compone de los siguientes pasos:

1. Se define el k a utilizar y se elijen los centroides iniciales, puede ser al azar.

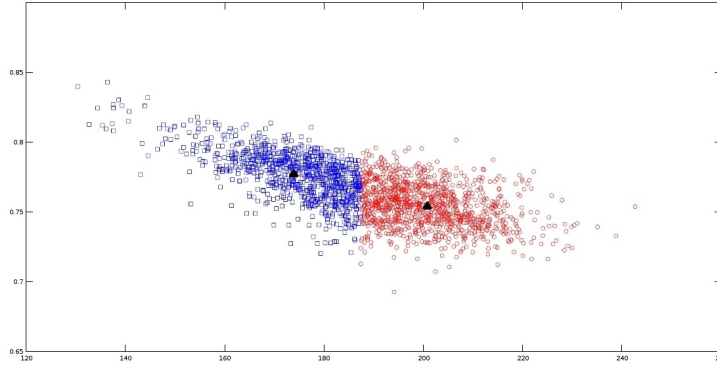


Figura 9: Ejemplo de particionamiento con el algoritmo k-means. Los puntos de color negro corresponden a los centroides de cada grupo.

2. Se calculan las distancias de cada objeto a los centroides y se asigna al más cercano.
3. Se calculan nuevamente los centroides.
4. Se repiten los pasos 2 y 3 hasta que los nuevos centroides calculados sean iguales a los anteriores.

En la Figura 9 se presenta un ejemplo de particionamiento en dos grupos ($k = 2$) para un conjunto de datos, donde los centroides para cada grupo se resaltan en color negro.

2.1.10 Support Vector Machine (SVM)

La máquina de soporte vectorial o SVM, construye un modelo basado en el análisis de datos y reconocimiento de patrones, este modelo es utilizado para clasificar nuevos elementos del tipo para el cual ha sido entrenado. De una manera más formal, para el caso de una clasificación binaria, si se tiene un conjunto de datos $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_{n-1}, y_{n-1}), (x_n, y_n)\}$, donde $x_i \in \mathbb{R}^d$, $y_i \in \{-1, +1\}$, se puede definir un hiperplano como una función lineal que es capaz de separar los datos en dos grupos [9]. De esta manera se plantea que el algoritmo SVM resuelve el problema de optimización de la Ecuación 12 donde w y b son coeficientes reales [35].

$$\min_{w,b} \quad \|w\|^2 + C \sum_{i=1}^n \epsilon_i \quad (12)$$

con las restricciones:

- $y_i(w^T x^{(i)} + b) \geq 1 - \epsilon_i$
- $\epsilon_i \geq 0$
- $i = 1, 2, 3, \dots, n$

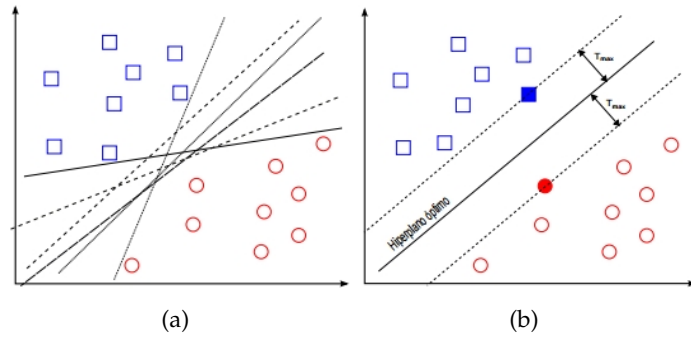


Figura 10: Hiperplanos lineales en un espacio bidimensional para un conjunto de datos separables en dos grupos. (a) Ejemplos de hiperplanos. (b) Hiperplano óptimo. Tomado de [9].

La Figura 10 muestra un ejemplo de hiperplanos lineales para un conjunto de datos separables en dos grupos. Intuitivamente se puede saber que hay infinitos hiperplanos lineales que pueden dividir un conjunto de datos (ver Figura 10(a)). Sin embargo, el hiperplano resultante de SVM será aquel que tenga la máxima distancia entre los datos más cercanos (ver Figura 10(b)).

La clasificación con SVM tiene tres grandes pasos:

- **Entrenamiento:** En este paso se construye el modelo de clasificación con base en datos etiquetados manualmente. Generalmente de un grupo de datos se toman el 80 % para realizar el entrenamiento del modelo.
- **Predicción:** En este paso se prueba el rendimiento del modelo, utilizando el 20 % de datos restantes del paso anterior, el modelo predice la etiqueta de cada elemento y con esto se determina su eficiencia.
- **Clasificación:** Una vez probado su rendimiento, utilizando nuevos datos, el modelo determina la categoría de cada elemento y, en el caso de la clasificación binaria, los separa en dos grupos.

2.2 TECNOLOGÍAS PARA EL DESARROLLO WEB MULTIMEDIA

2.2.1 HTML5

Es la última versión del lenguaje de hipertexto HTML. Este nuevo estándar provee características que se adaptan a las necesidades actuales de la Web, entre ellas se destacan las etiquetas de Audio y Video que facilitan la integración de contenido multimedia y algunas mejoras al canvas para el desarrollo 3D [48]. Dentro de sus novedades más importantes están:

- Etiquetas con códecs para mostrar contenido multimedia: audio, video, canvas 2D y 3D.
- Etiquetas para manejar grandes conjuntos de datos y que permiten generar tablas dinámicas que pueden filtrar, ordenar y ocultar contenido del lado del cliente.
- Nuevos tipos de datos como: *eMail*, *number*, *url*, *datetime*. Con los que se puede hacer validaciones fácilmente sin necesidad de Javascript.
- La funcionalidad *Drag & Drop* que permite arrastrar y soltar contenido.
- Incorpora nuevas etiquetas para manejar la Web Semántica, tales como: *header*, *footer*, *article*, *nav* o *time*.
- Permite la geolocalización.

2.2.2 Javascript

Javascript es un lenguaje ligero e interpretado que soporta funciones de primera clase. También es conocido como el lenguaje de scripts para páginas web [31]. Javascript se ejecuta en el lado del cliente. Con este lenguaje se puede diseñar e implementar el comportamiento de una página web y la ocurrencia de eventos.

La Figura 11(a) muestra un sencillo ejemplo de «Hola Mundo» utilizando Javascript.

2.2.3 JQuery

JQuery es una librería desarrollada para simplificar el uso de Javascript. Se caracteriza por ser rápida, liviana y tener implementadas una gran cantidad de funciones que hacen más fácil el uso de animaciones y AJAX [24].

La Figura 11b muestra un sencillo ejemplo de «Hola Mundo» utilizando jQuery. Se puede notar que se simplifica la sintaxis respecto al lenguaje javascript.

2.2.4 Bootstrap

Bootstrap es un potente framework *front-end* que combina HTML, Javascript y CSS para facilitar el desarrollo responsive en páginas web [7]. Este framework se ha convertido en el más usado por desarrolladores web ya que implementa las funciones más comunes en el diseño responsive como: tipografías, formularios, botones, tablas, barras de navegación, listas, *alerts* y *modals*.

1	<code><div id="my-div"></code>	HTML ⚙	1	CSS ⚙
2				
3	<code></div></code>			
1		JAVASCRIPT ⚙	Hola ESIC con Javascript	
2				
3	<code>document.getElementById("my-div").innerHTML = "Hola ESIC con Javascript";</code>			

(a)

1	<code><div id="my-div"></code>	HTML ⚙	1	CSS ⚙
2				
3	<code></div></code>			
1		JAVASCRIPT ⚙	Hola EISC con jQuery	
2				
3	<code>\$("#my-div").text("Hola EISC con jQuery");</code>			

(b)

Figura 11: “Hola Mundo” en HTML5 utilizando: (a) Javascript, (b) jQuery.
pruebas ejecutadas en jsfiddle.net

2.2.5 AJAX

Ajax es un conjunto de técnicas de desarrollo web que utilizan diferentes tecnologías del lado del cliente para crear aplicaciones web asíncronas. Ajax permite enviar y recibir datos del servidor y desplegarlos en pantalla sin interferir en la visualización actual de la página [16].

2.2.6 JW Player

Es un reproductor de videos web desarrollado en JavaScript que permite cambiar entre HTML5 o Flash para la reproducción de contenido cuando encuentra incompatibilidad en alguna de las dos tecnologías. Este reproductor se integra de manera eficiente con el servidor NGINX y su protocolo de transmisión RTMP. Además provee un API con todas las funcionalidades para su completa utilización [39].

2.2.7 FFmpeg

Es una herramienta de código abierto desarrollada en Linux pero que puede ser instalada en cualquier sistema operativo. Se usa por línea de comandos y sirve para convertir videos de un formato a otro. También provee la funcionalidad de obtener todos los fotogramas de cualquier video con múltiples opciones de extracción [15].

2.2.8 RTMP

Es un protocolo de transporte de contenido multimedia inicialmente desarrollado por Macromedia para transmitir videos entre un servidor y el reproductor de videos de Flash. Este protocolo se caracteriza por su eficiencia en la entrega de contenido multimedia a través de internet [3].

2.2.9 PHP

PHP es un lenguaje de código abierto muy utilizado para el desarrollo web y que puede ser incrustado en HTML. Generalmente sus scripts son desarrollados para la lógica de las aplicaciones web, sin embargo puede emplearse con múltiples propósitos. Lo que distingue a PHP de una tecnología del lado del cliente, como Javascript, es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente [18].

2.2.10 NGINX Server

Se trata de un servidor para aplicaciones web ligero y de alto rendimiento. Es de código abierto y se ejecuta sobre cualquier plataforma, en los últimos años se ha convertido en uno de los servidores más usados, superando a Microsoft Information Server [42]. Se compone de múltiples características, aunque la más importante para este trabajo de grado es su implementación del protocolo RTMP. Entre los servicios básicos que ofrece se encuentran:

- Servidor de archivos estáticos, índices y autoindexado.
- Proxy inverso con opciones de caché.
- Balanceo de carga.
- Tolerancia a fallos.
- Soporte de HTTP sobre SSL.
- Soporte para FastCGI con opciones de caché.
- Servidores virtuales basados en nombre y/o en dirección IP.
- Streaming de archivos FLV y MP4.8.
- Soporte para autenticación.
- Compatible con IPv6.
- Soporta más de 10.000 conexiones simultáneas.

2.2.11 PostgreSQL

PostgreSQL es un Sistema de gestión de bases de datos relacional orientado a objetos y de código abierto [17]. Posee múltiples ventajas y características, entre ellas:

- Alta concurrencia.
- Amplia variedad de tipos nativos.
- Llaves foráneas.
- Disparadores.
- Transacciones.
- Herencia de tablas.
- Funciones.

2.2.12 eXtreme Programming (XP)

La metodología de desarrollo de software eXtreme Programming, comúnmente denominada XP, es una metodología ágil propuesta por Kent Beck en el año 1999. Esta metodología se enfoca en la adaptabilidad del producto, el trabajo en equipo y la comunicación permanente con el cliente [13].

XP implementa, entre otros, los siguientes artefactos:

- **Historias de Usuario:** Es la forma en la que se presentan los requerimientos del sistema. Tienen el formato: *Actor - Acción - Propósito*. Por ejemplo: **Como Administrador quiero crear un nuevo empleado para agregarlo al sistema.**
- **Plan de entregas o Release Plan:** Es el cronograma de trabajo, en donde se especifican, por medio de iteraciones, la forma en que serán implementadas las historias de usuario.
- **Pruebas de Aceptación:** Son las pruebas funcionales que se realizan a las historias de usuario.

ESTADO DEL ARTE

A continuación se presentan algunos trabajos relacionados con este trabajo de grado. Se mostrarán ventajas y desventajas de cada uno y las razones por las que fue necesario realizar este proyecto.

3.1 CLASIFICACIÓN DE FOTOGRAMAS EN VIDEOS DE COLONOSCOPIA

En esta sección se presentan los trabajos más relevantes relacionados con la clasificación de fotogramas en videos de colonoscopia.

3.1.1 Clasificación Basada en Detección de Bordes

Esta clasificación se basa en el supuesto de que los fotogramas no informativos contiene una información distorcionada y con mucho ruido, contrario a los informativos, de los cuales se muestra un contenido mucho más claro y bien definido. Estos detalles se observan mejor cuando se extraen los bordes de las imágenes. La [Figura 12](#) presenta la comparación entre la información de los bordes de un fotograma informativo y otro no informativo.

Este método ha sido uno de los más estudiados y se pueden encontrar múltiples variantes en distintas publicaciones, normalmente el cambio está en el extractor de bordes y el manejo de los fotogramas ambiguos. Estos últimos son aquellos que, siendo no informativos son clasificados como informativos debido a su confusa información de los bordes (ver [Figura 13](#)). A continuación se muestra la versión original [\[34\]](#) y una variación de esta técnica [\[41\]](#).

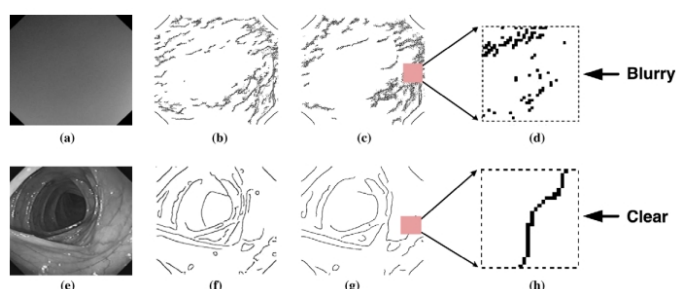


Figura 12: (a) Fotograma No Informativo, (b) y (c) bordes de (a), (d) detalles de los bordes de ruido, (e) Fotograma Informativo, (f) y (g) bordes de (e), y (h) detalle de los bordes limpios o bien definidos. Tomado de [\[34\]](#).

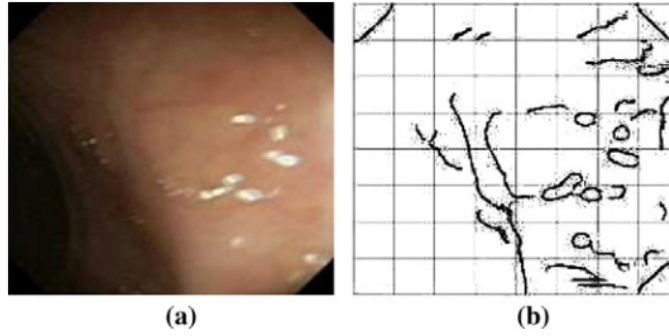


Figura 13: Fotograma Ambiguo (a) y sus bordes (b). Tomado de [34].

3.1.1.1 Utilizando el Detector de Bordes Canny

Este trabajo, presentado en [34], corresponde a la versión original del método por detección de bordes. Se utiliza el algoritmo Canny para extraer los bordes de cada fotograma y como medida de la información contenida se define el término *pixeles aislados*.

Los pixeles aislados son presentados como aquellos pixeles de borde que no están conectados con otros, es decir, una cadena de pixeles interconectados entre sí conforman un (1) *pixel aislado* o IP. Con el fin de calcular el porcentaje de IP de un fotograma, se propone la medida: *IPR*, que traduce: «razón de pixeles aislados» y se obtiene con la Ecuación 13.

$$IPR = \frac{\text{Total de Pixeles Aislados}}{\text{Total de pixeles del fotograma}} \times 100 \quad (13)$$

Finalmente, se establece una condición de clasificación donde se etiquetan como informativos aquellos fotogramas que tengan un IPR más bajo que un umbral y en caso contrario se etiquetan como no informativos.

El control de los fotogramas ambiguos se realiza dividiendo cada uno en 64 bloques y contando los bloques vacíos y los no vacíos. Seguidamente los bloques no vacíos son nuevamente clasificados entre ruidosos y limpios, de acuerdo a la medida de IPR. Finalmente, si un fotograma tiene más bloques limpios que ruidosos, se etiqueta como informativa.

La principal desventaja de esta forma de clasificación es que el detector de bordes Canny puede ser muy sensible a los umbrales T_{low} y T_{high} . La ventaja es la facilidad de implementación.

3.1.1.2 Utilizando el Operador Sobel con Filtros de Paso Alto y Bajo

El proceso realizado en el método de esta publicación [41] es similar a la anterior, excepto por que, en lugar de Canny, se utiliza el algoritmo de Sobel para extraer los bordes de los fotogramas. Adicio-

nalmente se refinan los bordes con operadores de paso alto y bajo con el fin de que estén mejor definidos, como aparecen en la [Figura 12](#).

También, se utiliza el término de pixel aislado, la medida IPR y la misma condición de clasificación de [34]. Sin embargo, para controlar los fotogramas ambiguos se utiliza un conteo general de pixeles individuales, donde se establece un nuevo umbral con el que se define si el fotograma es informativo en caso de que tenga un valor de pixeles mayor que el umbral, o no informativo en el caso contrario.

La ventaja de este método, a parte de que es fácil de implementar, es no son necesarios los umbrales (T_{low} y T_{high}) del algoritmo Canny. Pero su desventaja está en que los filtros de paso alto y bajo utilizados son sensibles a los valores en las ventanas de convolución.

3.1.1.3 Segmentación de Brillo para Mejorar la Clasificación por Detección de Bordes

Como se puede observar en la [Figura 13](#), el fotograma presenta algunas zonas blancas que forman nuevos bordes y produce una mala clasificación. Estas zonas son conocidas como *zonas de brillo* y se deben a la reflexión entre el rayo de luz que sale del colonoscopio y el agua que es utilizada para limpiar las paredes del colon. Las zonas de brillo representan un problema para la clasificación por detección de bordes, dado que, se generan *falsos bordes* que son difíciles de identificar para el algoritmo de clasificación.

En [34], se propone un método de segmentación y eliminación de zonas de brillo con el fin de reducir el número de fotogramas ambiguos y mejorar la clasificación por detección de bordes. La segmentación consiste, a grandes rasgos, de tres pasos:

- Se convierte la el fotograma del espacio de color [RGB](#) a [HSV](#).
- Se definen las areas de brillo utilizando 2 umbrales: Th_s para la saturación (S) y Th_v para el Value (V). Se define una condición para cada pixel i , donde se marca como brillo cuando $S(i) < Th_s$ y $V(i) > Th_v$ y en caso contrario se descarta como pixel de brillo.
- Se calculan los bordes de la imagen con las zonas de brillo segmentadas con el fin de eliminarlos de los bordes del fotograma original.

La [Figura 14](#) muestra el resultado de eliminar los falsos bordes de los fotogramas de colonoscopia.

La utilización de esta técnica mejora la clasificación por detección de bordes, ya que reduce el número de bordes de brillo que no hacen parte de la verdadera información del fotograma.

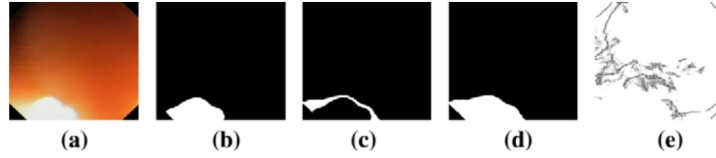


Figura 14: (a) Fotograma no informativo con zonas de brillo, (b) Área de brillo absoluta de (a), (c) Área relativa de brillo de (a), (d) Mapa de reflexión de brillo de (a), y (e) mejoramiento de bordes de (a) eliminando los bordes de las zonas de brillo. Tomado de [34].

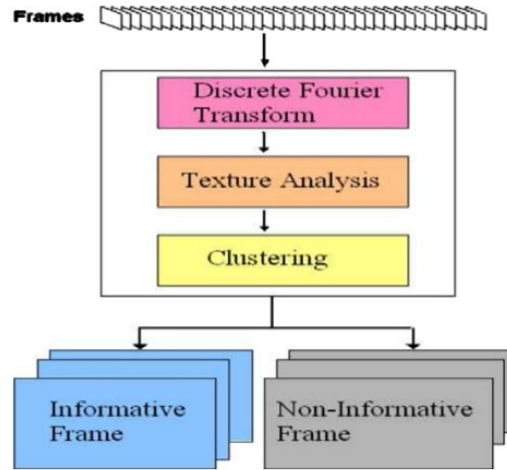


Figura 15: Proceso general de la técnica de clasificación por patrones de textura propuesta en [34]

3.1.2 Clasificación Basada en Patrones de Textura

En [34] se incluye una técnica de clasificación de fotogramas basada en patrones de textura. La idea principal es realizar la transformada de Fourier a los fotogramas de colonoscopia y analizar la textura en el dominio de sus frecuencias. Finalmente, con descriptores de textura se realiza un agrupamiento utilizando k-means. El proceso general del método se muestra en la Figura 15.

En este método se identifica que los fotogramas informativos presentan patrones de textura diferentes a los no informativos en el dominio de las frecuencias, tal como se puede observar en la Figura 16.

Para extraer características de texturas son utilizadas las métricas de Haralick sobre la matriz de co-ocurrencias del espectro de la transformada de Fourier del fotograma. Por cada fotograma se obtiene un vector de 7 posiciones, que es tomado como descriptor, donde cada posición corresponde a un valor de cada medida de Haralick.

Finalmente se utiliza el algoritmo k-means para clasificar los fotogramas en informativos y no informativos.

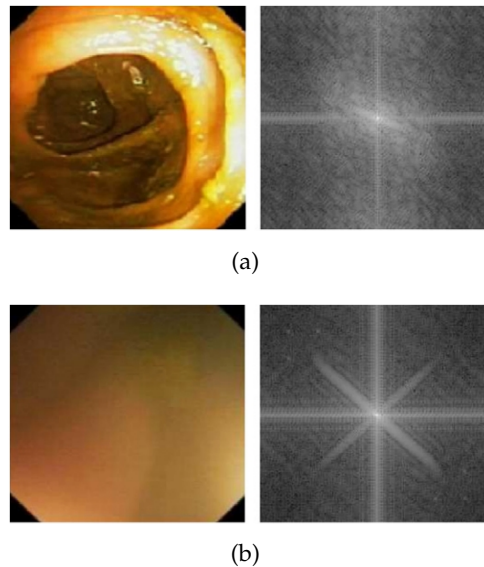


Figura 16: (a) Fotograma Informativo y su representación en el dominio de Fourier. (b) Fotograma No Informativo y su representación en el dominio de Fourier. Tomado de [34].

3.1.3 Otros Métodos de Clasificación

3.1.3.1 Clasificación de Fotogramas de Colonoscopia en Tiempo Real con Unidades de Procesamiento Gráfico (GPU)

El trabajo realizado en [25] corresponde a la tesis de maestría del estudiante KarriVenkata, de la Universidad del Norte de Texas, quien propone métodos para clasificar fotogramas informativos y no informativos de los videos de colonoscopia utilizando múltiples hilos de procesamiento de máquina (CPU) y gráficos (GPU). El método para el reconocimiento es el mismo que propone [34] con el componente adicional del procesamiento paralelo para realizar la clasificación en tiempo real. Esto es una ventaja para el método de detección de bordes con Canny, sin embargo sigue siendo necesario el control sobre los umbrales del algoritmo detector de bordes.

3.1.3.2 Clasificación de Fotogramas Basada en Medidas de Luminancia

En [49] se propone una clasificación basada en detección de luz ponderada. Los fotogramas son seleccionados de acuerdo a su percepción de profundidad, determinada por la cantidad de luz total. La medida de luminancia o *lumen* es calculada con la información proporcionada por la técnica de análisis de características espaciales: *Mean Shift*, el cual segmenta el fotograma con base en los puntos, en un espacio de color, que corresponden a la probabilidad de una función de densidad. Con la segmentación realizada por *Mean Shift* se clasifican los fotogramas en informativos y no informativos.

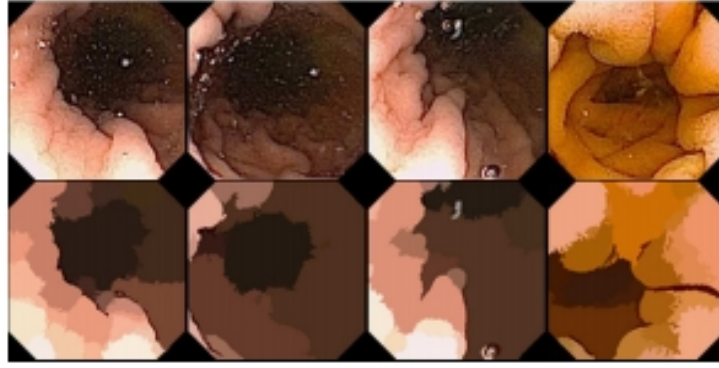


Figura 17: Primera fila: Fotogramas originales obtenidos con una cápsula inalámbrica. Segunda fila: Fotogramas segmentados con *Mean Shift*. Tomado de [49].

Los pasos del método propuesto son los siguientes:

- Se convierte cada fotograma del espacio de color RGB a HSI.
- Los fotogramas son segmentados con el algoritmo *Mean Shift*, como lo muestra la Figura 17.
- Se calcula la medida de luminancia para cada fotograma S_i^L con la Ecuación 14.

$$S_i^L = I^2 \times \frac{A_L}{A} \quad (14)$$

donde: I es la intensidad promedio normalizada. A es el área total del fotograma y A_L es el área total iluminada.

- Si el *lumen* obtenido para un fotograma es más bajo que un umbral pre-definido, se elimina. En caso contrario se trata de un fotograma informativo y se conserva.

3.2 HERRAMIENTAS PARA ANOTACIÓN DE VIDEOS

3.2.1 Open Video Annotation Project (OVA)

Se trata de un proyecto web para la anotación de videos, desarrollado con librerías JavaScript de código abierto y utilizando el nuevo estándar HTML5. Esta aplicación propone anotaciones sobre la barra de progreso del video, las cuales corresponden a una franja de color amarillo que despliega un “pop-up” con la respectiva anotación que puede ser en texto, imágenes o incluso un video [44]. Una de sus principales ventajas, además de la posibilidad de insertar imágenes o videos, es que las anotaciones pueden ser redactadas en texto con formato, se pueden añadir hipervínculos y la ubicación del anotador. Su



Figura 18: Interfaz gráfica de OVA

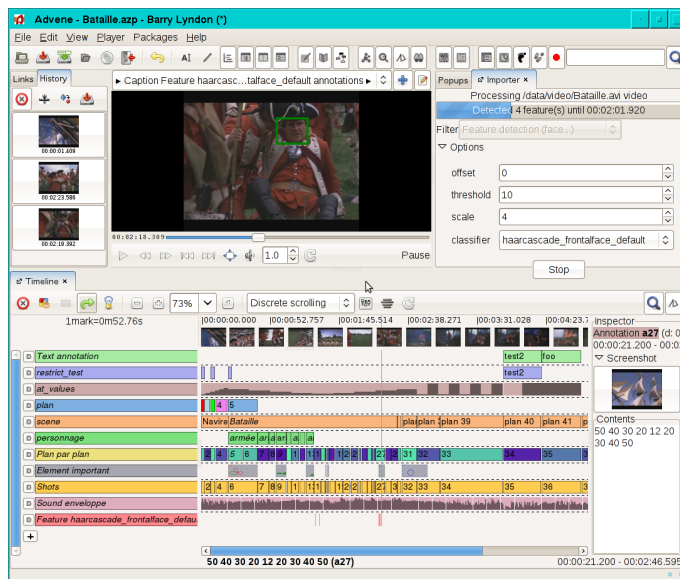


Figura 19: Interfaz gráfica de Advène

desventaja es que aún no está disponible para uso público. La [Figura 18](#) muestra una captura de la interfaz gráfica principal del proyecto OVA.

3.2.2 Advène Project

Advène [5] es una herramienta de anotación de videos de la Universidad Claude Bernard Lyon 1 en Francia. Aunque esta aplicación aparece aún en fase de desarrollo se puede obtener de manera gratuita un instalador para cualquier plataforma (Windows, Linux, Mac). Cuenta con un amplio conjunto de funcionalidades que apoyan la anotación y análisis detallado de videos. Entre sus objetivos está el intercambio de anotaciones. Su principal ventaja es que se puede visualizar en una línea de tiempo todos los fotogramas facilitando el proceso de anotación. La desventaja es su compleja manipulación debido a todas las funcionalidades que posee, teniendo en cuenta que,

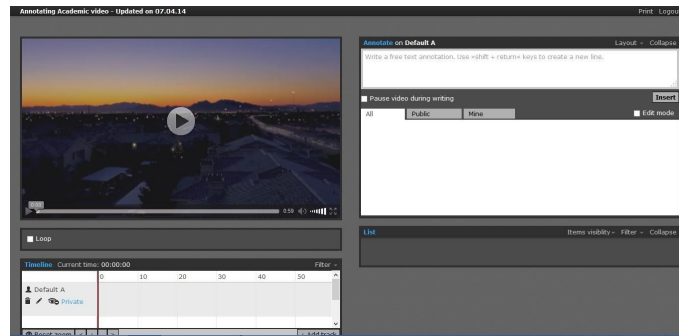


Figura 20: Interfaz gráfica de Anotating Academic Video

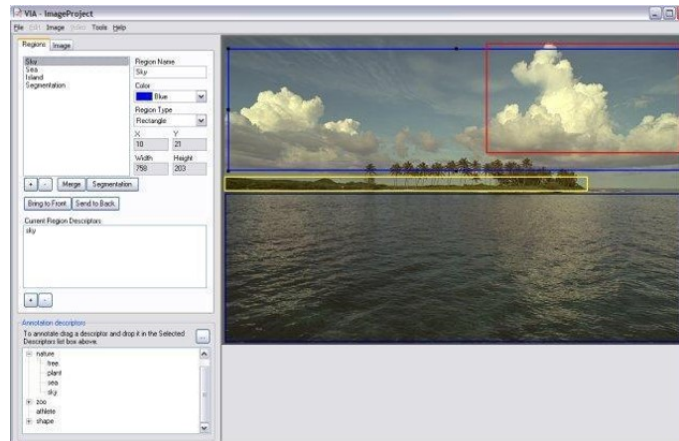


Figura 21: Interfaz gráfica de VIA

en este trabajo de grado, quienes realizarán las anotaciones serán los médicos, los generalmente requieren aplicaciones prácticas y de fácil uso. La [Figura 19](#) presenta una captura de la interfaz gráfica principal de esta aplicación.

3.2.3 Anotating Academic Video (AAV)

AAV [43] es un proyecto de código abierto desarrollado en HTML5 y JavaScript que permite realizar anotaciones de videos en formato de texto plano, provee funcionalidades muy útiles de reproducción y su línea de tiempo hace bastante fácil la tarea de recorrer el video para realizar las anotaciones. Una ventaja de esta herramienta es que las anotaciones se pueden compartir de manera pública o privada y su interfaz es sencilla de manejar. La desventaja es que está en proceso de desarrollo y la demostración disponible en línea no permite subir videos propios. La [Figura 20](#) ilustra la interfaz gráfica principal de la aplicación AAV.

3.2.4 *Video Image Annotation Tool (VIA)*

VIA [45] es una aplicación diseñada para Windows en la que se pueden realizar anotaciones a imágenes y videos, su interfaz gráfica es bastante sencilla. Su ventaja principal es que controla la reproducción y anotación del video con los fotogramas, esto permite que sea más sencillo el proceso de anotación. La desventaja es que solo está disponible para Windows. En la [Figura 21](#) se presenta una captura de la interfaz gráfica principal de la aplicación VIA.

CLASIFICACIÓN AUTOMÁTICA DE FOTOGRAMAS EN VIDEOS DE COLONOSCOPIA

El propósito de la clasificación de fotogramas en videos de colonoscopia es distinguir aquellos que son informativos de los que son no informativos. A continuación, se presenta la definición de informativo y no informativo, para los fotogramas que serán usados en este trabajo de grado, y en la [Figura 22](#) se muestran algunos ejemplos:

- **Fotograma Informativo:** El contenido de la imagen es claro y está bien definido.
- **Fotograma No Informativo:** El contenido de la imagen aparece desenfocado, borroso o con demasiado brillo en algunas zonas.

Para el reconocimiento y descarte automático de fotogramas no informativos en videos de colonoscopia, se proponen dos métodos: el primero basado en detección de bordes y el segundo basado en descriptores de textura. Los resultados de la clasificación basada en bordes ([Sección 4.1](#)) fueron presentados en la conferencia **LACNEM-2015** (www.lacnem.org), realizada en la Facultad de Minas de la Universidad Nacional en Medellín, con el artículo: *Automatic Classification of Non-Informative Frames in Colonoscopy Videos* [8] y se espera presentar los resultados del método de textura ([Sección 4.2](#)) en el próximo congreso colombiano de computación **11ccc** (fup.edu.co/11ccc) que se realizará en la ciudad de Popayán entre el 27 y el 30 de Septiembre de 2016.

A continuación se describen los métodos propuestos con sus respectivas evaluaciones.

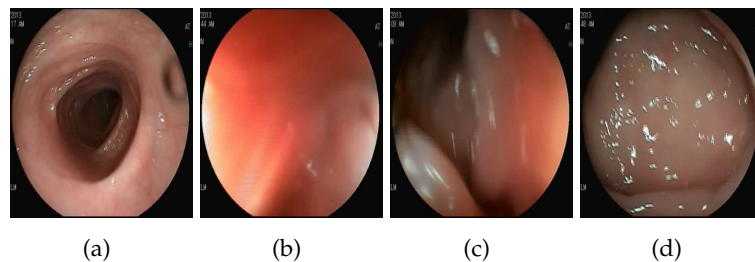


Figura 22: Ejemplos de fotogramas en videos de colonoscopia: (a) Fotograma Informativo. (b) Fotograma No Informativo debido al contacto de la cámara con las paredes del colon. (c) Fotograma No Informativo por borrosidad. (d) Fotograma No Informativo por reflexión de la luz con el agua.

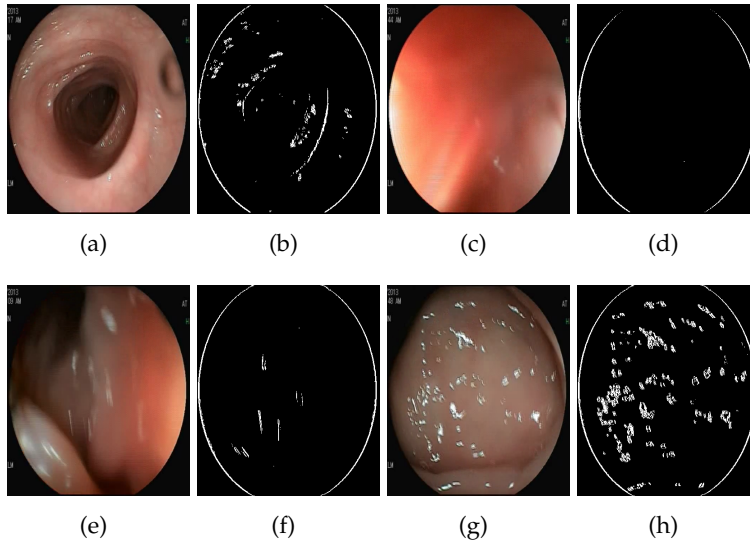


Figura 23: Bordes en fotogramas de videos de colonoscopia: (a) Fotograma Informativo. (b) Bordes de (a). (c) Fotograma No Informativo debido al contacto de la cámara con las paredes del colon. (d) Bordes de (c). (e) Fotograma No Informativo por borrosidad. (f) Bordes de (e). (g) Fotograma No Informativo por reflexión de la luz con el agua. (h) Bordes de (g).

4.1 CLASIFICACIÓN BASADA EN DETECCIÓN DE BORDES

Los bordes son un indicador de la cantidad de contenido en una imagen. Normalmente, un fotograma informativo contiene más bordes que un fotograma no informativo. La [Figura 23](#) muestra algunos ejemplos de fotogramas con sus respectivos bordes obtenidos con el operador Sobel.

La técnica de clasificación de fotogramas de colonoscopia basada en detección de bordes fue propuesta inicialmente en [\[34\]](#) usando el detector de bordes Canny. En [\[41\]](#) se propone una variante de este método utilizando el operador Sobel con filtros de paso alto y filtros de paso bajo. Ambos métodos son presentados en la [Subsección 3.1.1](#).

En este trabajo de grado se propone un método basado en detección de bordes usando el operador Sobel, y se complementa con una técnica de segmentación de brillo con el fin de refinar la clasificación.

La [Figura 24](#) ilustra el diagrama general del método propuesto. En primer lugar, se extraen todos los fotogramas del video de colonoscopia utilizando la librería FFMPEG [\[15\]](#). Como segundo paso se utiliza la clasificación por detección de bordes, descrita en la [Subsección 4.1.1](#), para dividir los fotogramas en dos grupos: *Fotogramas Ambiguos* y *Fotogramas No Informativos*. El primer grupo contiene fotogramas informativos y no informativos. Finalmente, se emplea la técnica de segmentación de brillo que se muestra en la [Subsección 4.1.2](#)

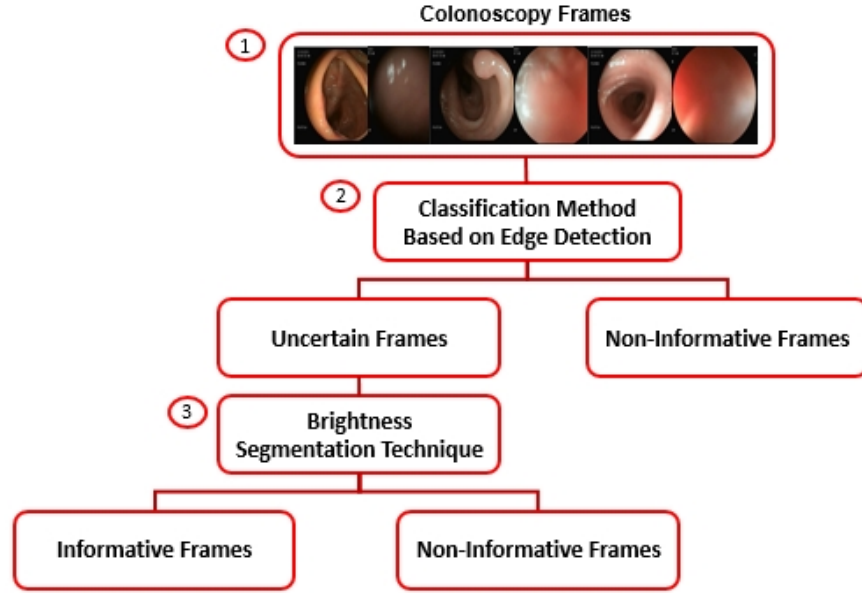


Figura 24: Diagrama general del método propuesto basado en detección de bordes

para separar, del grupo de fotogramas ambiguos, los fotogramas que son informativos de los que son no informativos.

4.1.1 Clasificación por Detección de Bordes

Dado que los bordes pueden mostrar el volumen de contenido en una imagen; se asume que un fotograma informativo tiene un mayor número de bordes que un no informativo. Con el fin de calcular el porcentaje de bordes o pixeles de bordes presentes en un fotograma, se define la medida Edge Pixel Percentage (EPP) en la Ecuación 15.

$$EPP = \frac{\text{Numero de pixeles de borde}}{\text{Total de pixeles}} \times 100. \quad (15)$$

La Figura 25 ilustra el proceso de la clasificación por detección de bordes. Como primer paso se obtienen los bordes del fotograma original con el operador Sobel, luego se calcula el porcentaje de pixeles de bordes utilizando la Ecuación 15. Finalmente, se compara si el EPP resultante es mayor que un umbral Th establecido previamente. Si la condición es verdadera el fotograma es etiquetado como *ambiguo*, en caso contrario el se etiqueta como *no informativo*.

En el Algoritmo 1 se presenta el pseudocódigo de la clasificación por detección de bordes y en el Algoritmo 2 se ilustra el proceso para el cálculo del porcentaje de pixeles de bordes en un fotograma.

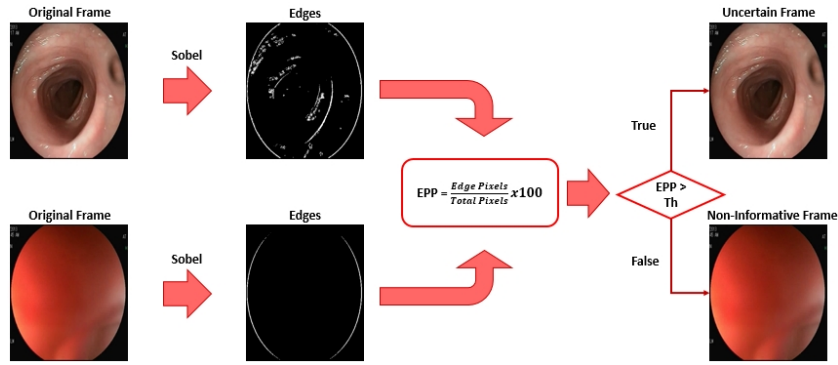


Figura 25: Proceso de la clasificación por detección de bordes

Algoritmo 1 Clasificación de fotogramas por detección de bordes**Entrada:** Lista de fotogramas, Umbral (Th)**Salida:** Dos listas: fotogramas *ambiguos* y *no informativos*
para todo Fotograma F en ListaFotogramas **hacer** $sobelImg \leftarrow Sobel(F)$ $EPP \leftarrow calcularPixelesBordes(sobelImg)$ **si** $EPP \geq Th$ **entonces** ListaAmbiguos $+= F$ **si no** ListaNoInformativos $+= F$ **fin si****fin para****Algoritmo 2** Cálculo del porcentaje de pixeles de borde en una imagen**Entrada:** Imagen de bordes binaria**Salida:** Porcentaje de pixeles de borde en la imagen $pixelesBlancos = 0, pb = 0$ **para** $i = 0$ hasta $i = anchoImagen$ **hacer** **para** $j = 0$ hasta $j = altoImagen$ **hacer** **si** $imagen[i][j] == 255$ **entonces** $pixelesBlancos ++$ **fin si** **fin para** **fin para** $pb = \frac{pixelesBlancos}{anchoImagen \times altoImagen} \times 100$ **devolver** pb

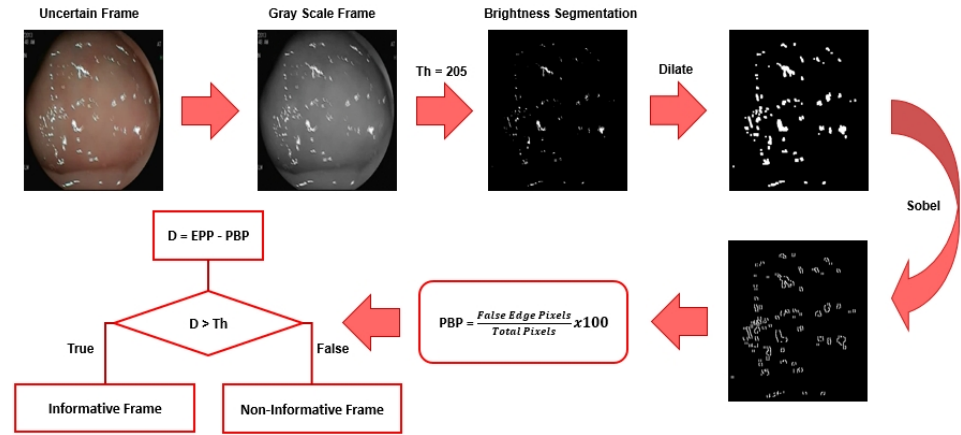


Figura 26: Técnica de segmentación de brillo

4.1.2 Segmentación de Brillo

La reflexión de la luz en el agua que se utiliza para limpiar las paredes del colon, produce falsos bordes que afectan negativamente la clasificación de los fotogramas, dado que el clasificador los puede etiquetar como informativos y un fotograma con demasiadas zonas de brillo es tomada por los especialistas como no informativo. En [34] se realiza una segmentación del brillo en los fotogramas, basados en el espacio de color *HSV* (ver la Subsubsección 3.1.1.3 del Capítulo 3).

De acuerdo con lo anterior, se propone una técnica de segmentación de brillo que permite identificar los falsos bordes y mejorar la clasificación. Como se muestra en la Figura 26, la técnica de segmentación de brillo se compone de los siguientes pasos: El fotograma es convertido del espacio de color *RGB* a escala de grises para obtener una imagen binaria aplicando un umbral con valor 205. El valor del umbral es calculado experimentalmente con base en la información suministrada por los histogramas de los fotogramas en escala de grises, en la Figura 27 se puede observar que el brillo corresponde a los valores más altos del histograma.

Los pixeles de brillo en la imagen binaria obtenida de la umbralización son modificados morfológicamente, usando la operación de dilatación con un elemento estructural rectangular de 5×5 . Esto se realiza con el fin de aumentar el volumen de los pixeles de brillo para extraer mejor sus características. El siguiente paso es obtener los bordes de las zonas de brillo con el operador Sobel. Posteriormente, para calcular el porcentaje de pixeles de brillo en la imagen, se propone la medida: Pixel Brightness Percentage (*PBP*) (Ecuación 16) cuyo cálculo se realiza empleando el mismo algoritmo con que se calcula el *EPP*.

$$PBP = \frac{\text{Numero de pixeles de brillo}}{\text{Total de pixeles}} \times 100. \quad (16)$$

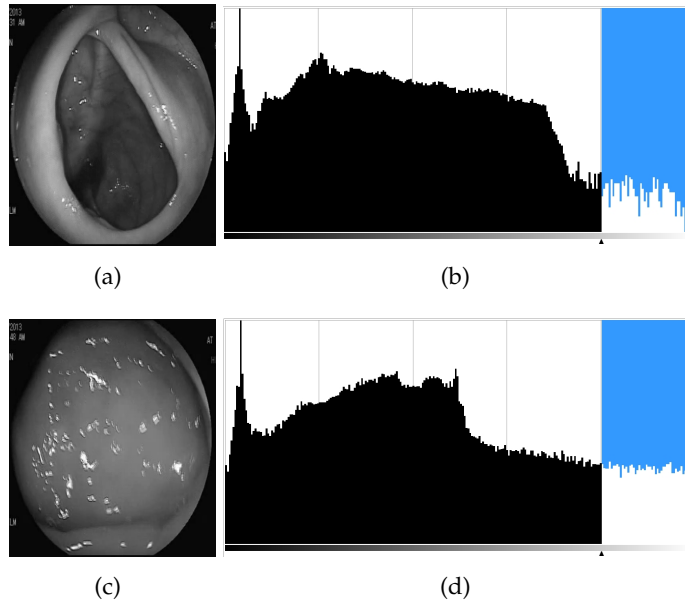


Figura 27: Análisis de brillo en los fotogramas. (a) Fotograma Informativo con pocas zonas de brillo. (b) Histograma de (a). (c) Fotograma no informativo con altas zonas de brillo. (d) Histograma de (c). El rango entre 205 y 255 se muestra en color blanco con fondo azul, resaltando las frecuencias donde se concentra el brillo en las imágenes

A continuación, se obtiene el **PBP** de la imagen y se calcula la diferencia (**D**) con su respectivo **EPP**, como se indica en la **Ecuación 17**

$$D = EPP - PBP \quad (17)$$

Finalmente, se compara la diferencia **D** obtenida con el umbral **Th** definido al principio. Si **D** es mayor que **Th**, indica que la imagen no contenía muchas zonas de brillo, por lo tanto el fotograma es etiquetado como informativo. En caso contrario, los bordes de la imagen eran, en gran parte, falsos bordes producidos por la reflexión de la luz, por lo tanto el fotograma es etiquetado como no informativo.

El Algoritmo 3 muestra la técnica de segmentación de brillo.

4.2 CLASIFICACIÓN BASADA EN DESCRIPTORES DE TEXTURA

La textura provee información importante acerca de la distribución espacial de elementos fundamentales de una imagen [2]. Actualmente es posible caracterizar una imagen de acuerdo al análisis de textura utilizando diferentes medidas o descriptores. En el caso de los fotogramas de colonoscopia se han utilizado las *Métricas de Haralick* como descriptores de textura obteniendo buenos resultados, tal como se muestra en la **Subsección 3.1.2**.

Algoritmo 3 Técnica de segmentación de brillo**Entrada:** Lista de fotogramas *ambiguos*, Umbral (Th)**Salida:** Dos listas: fotogramas *informativos* y *no informativos***para todo** Fotograma F en ListaFotogramasAmbiguos **hacer** $grayScaleImg \leftarrow RGB2GRAY(F)$ $binaryImg \leftarrow aplicarUmbral(grayScaleImg, 205)$ $dilateImg \leftarrow aplicarDilatacion(binaryImg)$ $sobelImg \leftarrow Sobel(F)$ $PBP \leftarrow calcularPixelesBordes(sobelImg)$ $D \leftarrow F.EPP - PBP$ **si** $D \geq Th$ **entonces** $ListaInformativos += F$ **si no** $ListaNoInformativos += F$ **fin si****fin para**

En este trabajo de grado se propone una modificación al método de clasificación de fotogramas de colonoscopia basado en descriptores de textura presentado en [34]. La modificación consiste en utilizar el operador LBP en lugar de las *Métricas de Haralick*. Además se clasifican los fotogramas de los videos utilizando tanto el algoritmo *k-means* como el clasificador SVM.

A continuación se presentan las etapas de la clasificación basada en descriptores de textura: preprocesamiento de los fotogramas, análisis de textura y finalmente la clasificación con k-means y SVM.

4.2.1 Preprocesamiento

La Transformada de Fourier DFT es una importante herramienta que se utiliza para entender la naturaleza de una imagen y su información espacial [2]. Con el fin de obtener mejores características de los fotogramas de colonoscopia, se convierte cada uno a escala de grises y se obtiene su representación en el dominio de Fourier. En esta nueva imagen es posible identificar patrones que pueden ser analizados con la ayuda de diferentes descriptores. De acuerdo con la Figura 28, es posible observar que, según el contenido de los fotogramas, en las imágenes de la Transformada Discreta de Fourier (DFT) se generan diferentes patrones. Por ejemplo, en Figura 28(d) se observan líneas diagonales que salen del centro de la imagen, mientras que en Figura 28(b) se ve una distribución uniforme entre pixeles blancos y oscuros.

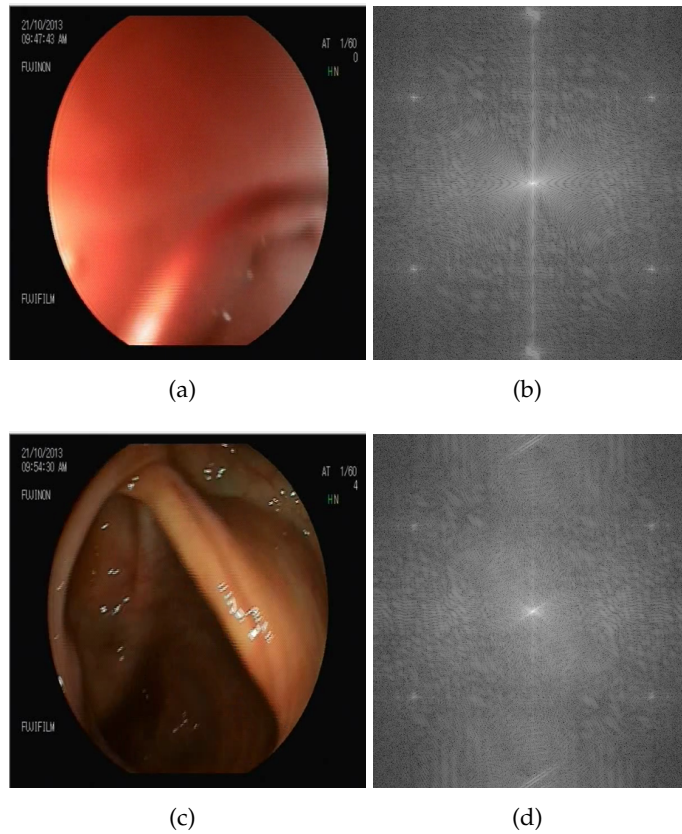


Figura 28: (a) Fotograma No Informativo. (b) Espectro de Frecuencias de (a). (c) Fotograma Informativo. (d) Espectro de Frecuencias de (c)

4.2.2 Análisis de Textura

El análisis de textura se realiza en el espectro de las frecuencias de cada imagen con el fin de identificar mejores patrones que caractericen y diferencien los fotogramas informativos de los no informativos. **LBP** es uno de los descriptores más usados para el análisis de textura, además posee ventajas como la invarianza frente a cambios en la imagen como iluminación o desenfoque y un bajo costo computacional, que permite realizar análisis de imágenes en tiempo real [38].

Para el desarrollo de este método se utilizó el código para Matlab desarrollado en [20]. El descriptor **LBP** modifica cada pixel con base en una umbralización realizada entre el pixel de interés y sus vecinos. Finalmente, se obtiene un vector de frecuencias de intensidades con el que es posible caracterizar una imagen, tal como se describe en la Subsubsección 2.1.8.2. El vector obtenido por cada imagen es utilizado para clasificar los fotogramas en informativos y no informativos empleando el algoritmo k-means o **SVM**.

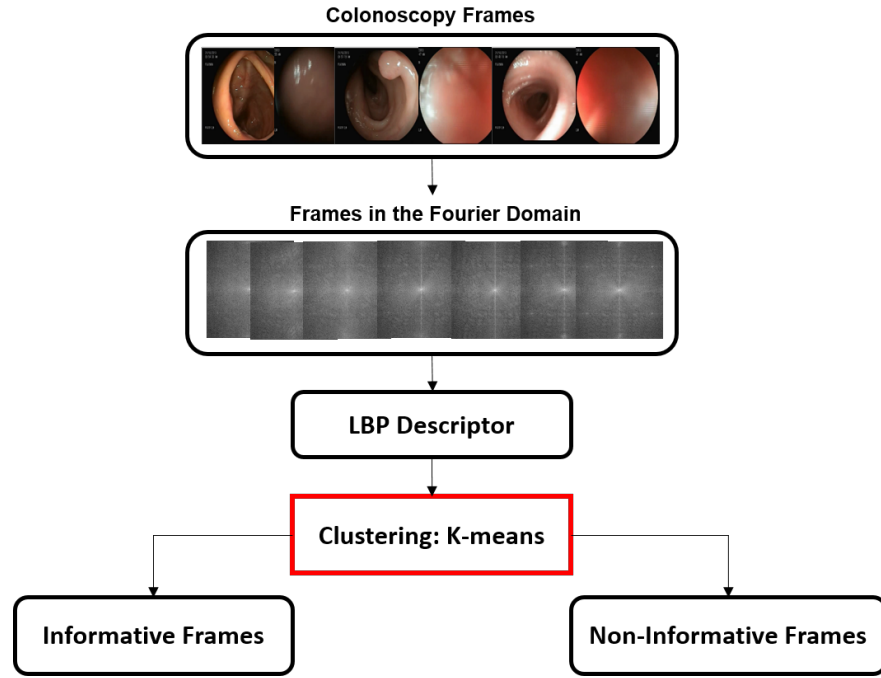


Figura 29: Diagrama general de la clasificación de los fotogramas de colonoscopia con *k-means*

4.2.3 Clasificación de Fotogramas con *k-means*

El algoritmo de particionamiento *k-means* divide un conjunto de entrada en *k* grupos, donde *k* se define al inicio del mismo [2]. El algoritmo consta de 4 pasos que se describen en la Subsección 2.1.9. Para realizar el particionamiento, en este trabajo establece un $k = 2$ y se construye una matriz de clasificación, que sirve como datos de entrada del algoritmo, con el mismo número de columnas del descriptor y con tantas filas como fotogramas se tengan, es decir, en el caso de que el vector del descriptor tenga 256 posiciones y se usen 600 fotogramas, la matriz de clasificación será de tamaño 256×600 .

La Figura 29 muestra el diagrama general de la clasificación de fotogramas de colonoscopia con *k-means*. Inicialmente, se extraen todos los fotogramas del video con la librería FFMPEG [15]. Luego, en la etapa de preprocesamiento, se obtiene una imagen del espectro de la DFT por cada fotograma. El siguiente paso es analizar la textura y caracterizar cada imagen del dominio de las frecuencias con el descriptor LBP. Finalmente, utilizando la implementación en Octave del algoritmo *k-means*, se forman 2 grupos: *Fotogramas Informativos* y *Fotogramas No Informativos*.

4.2.4 Clasificación de Fotogramas con SVM

La clasificación de fotogramas de colonoscopia con base en el análisis de textura y SVM se realiza utilizando la librería LIBSVM [10] con los siguientes parámetros:

- Una matriz de características, que se construye con todos los descriptores LBP de cada fotograma. El tamaño de esta matriz ($M \times N$) es determinado por la longitud del descriptor N , y el número total de fotogramas utilizados M .
- Un vector (columna) de etiquetas, que contiene en cada posición un valor que identifica el tipo de fotograma de cada fila de la matriz de características: 1 para los fotogramas informativos y -1 para los no informativos.
- Una cadena de ajustes, que cambia dependiendo del uso que se le quiera dar al algoritmo. En este caso se utiliza la cadena: $'-s 0 -t 0 -c 100'$. Donde: $-s 0$ hace referencia a la clasificación con SVM, $-t 0$ indica que se debe utilizar un kernel lineal y $-c 100$ es el costo que de la variable c , un costo pequeño, como $c 1$, es menos sensible a los errores de clasificación que un costo más grande.

4.3 EVALUACIÓN DE LAS TÉCNICAS

En esta sección se evalúan los métodos propuestos en las dos secciones anteriores. A continuación se establecen un conjunto de datos, se describen las medidas de desempeño y se proponen condiciones de evaluación para cada método. Al final, se realiza un análisis y comparación entre los resultados de ambas técnicas. Todas las pruebas fueron ejecutadas en un PC Laptop con Windows 8 Pro x64, procesador Intel (R) Core (TM) i5 @ 2.60 GHz y 4,00 GB de memoria RAM.

4.3.1 Conjunto de Datos

El rendimiento de los métodos propuestos es evaluado usando tres videos de colonoscopia, los cuales han sido adquiridos por el Dr. Jose Arbey Hoyos del Hospital Universitario del Valle. Cada video tiene una duración de 6,12 y 14 minutos respectivamente, la resolución es de 636×480 pixeles y han sido grabados a una tasa de 10 fps usando el formato MP4 y H.264 como algoritmo de compresión.

De cada video fueron seleccionados 100 fotogramas informativos y 100 no informativos en formato PNG, para un conjunto total de 600 fotogramas. La selección se realizó con la ayuda de especialistas médicos, quienes utilizaron una herramienta web que se desarrolló para este propósito específico. La aplicación permite clasificar por medio

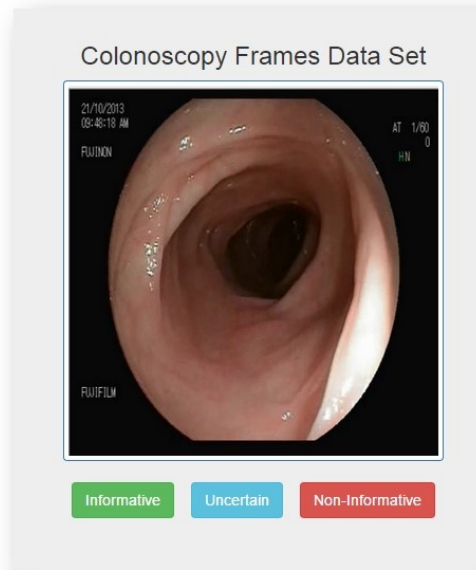


Figura 30: Interfaz gráfica de la aplicación web para la selección de los fotogramas.

de tres opciones cada fotograma de un video. En la [Figura 30](#) se presenta la interfaz gráfica de la aplicación. Básicamente, el especialista oprime el botón verde si considera que el fotograma es informativo, el rojo si se trata de un fotograma no informativo o el azul en caso de que la imagen sea confusa.

4.3.2 Medidas de Desempeño

Para evaluar el desempeño de los métodos se emplea un conjunto de medidas comunmente utilizadas en la evaluación de algoritmos de clasificación binaria [40].

Antes de definir las métricas, es necesario establecer algunos términos que estas utilizan y con los que más adelante se construye la matriz de confusión.

Los términos son los siguientes:

- *True Informative* (TI): Fotograma Informativo correctamente clasificado,
- *False Informative* (FI): Fotograma No Informativo incorrectamente clasificado como Informativo,
- *True Non-Informative* (TNI): Fotograma No Informativo correctamente clasificado, y
- *False Non-Informative* (FNI): Fotograma Informativo incorrectamente clasificado como No Informativo.

Las medidas seleccionadas para la evaluación del rendimiento son las siguientes:

- *Sensibilidad* (Sen): Mide la correcta clasificación de fotogramas como Informativos. Esta medida se define como:

$$\text{Sen} = \frac{\sum \text{TI}}{\sum \text{TI} + \sum \text{FNI}} \quad (18)$$

- *Especificidad* (Spec): Mide la correcta clasificación de los fotogramas como No Informativos. Esta medida se define como:

$$\text{Spec} = \frac{\sum \text{TNI}}{\sum \text{TNI} + \sum \text{FI}} \quad (19)$$

- *Exactitud* (Acc): Calcula el porcentaje de fotogramas correctamente clasificados como Informativos y No Informativos. Esta medida se define como:

$$\text{Acc} = \frac{\sum \text{TI} + \sum \text{TNI}}{\sum \text{TI} + \sum \text{FI} + \sum \text{TNI} + \sum \text{FNI}} \quad (20)$$

- *Precisión* (Prec): Calcula la razón entre el número de fotogramas correctamente clasificados como Informativos y el total del número de fotogramas correctamente clasificados como informativos y el número de fotogramas incorrectamente clasificados como Informativos. Esta medida se define como:

$$\text{Prec} = \frac{\sum \text{TI}}{\sum \text{TI} + \sum \text{FI}} \quad (21)$$

- *F-Measure* (FM): Calcula la media armónica entre la precisión y la sensibilidad. El valor ideal para esta medida es 1 y se define como:

$$\text{FM} = 2 \cdot \frac{\text{Prec} \cdot \text{Sen}}{\text{Prec} + \text{Sen}} \quad (22)$$

4.3.3 Evaluación del Método de Detección de Bordes

La evaluación del método de detección de bordes, presentado en la [Sección 4.1](#), se realiza en tres escenarios diferentes: Primero se clasifican los fotogramas sin la técnica de segmentación de brillo y luego con ella. El umbral Th establecido para la decisión de clasificación fue de 2,5. Además, se realiza una comparación con el método basado en detección de bordes con Canny propuesto en [34]. El criterio de decisión para el método de Canny se basó en el supuesto de que un fotograma informativo tiene más bordes aislados que uno no informativo. El umbral utilizado en la clasificación con Canny fue de 0,08 y los umbrales alto y bajo para el detector de bordes fueron de 20 y 50 respectivamente.

En la [Tabla 1](#) se presentan los resultados de la clasificación de fotogramas, en cada uno de los tres escenarios de evaluación propuestos para el método basado en detección de bordes: Con y sin la técnica de segmentación de brillo y el método de Canny. En la [Tabla 2](#) se presentan los resultados de las medidas de rendimiento.

Tabla 1: Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en detección de bordes con y sin la técnica de segmentación de brillo y el método de Canny propuesto en [34]

Clasificación	TI	FI	TNI	FNI
Sin segmentación de brillo	241	99	201	59
Con segmentación de brillo	240	88	212	60
Canny	288	88	212	12

Tabla 2: Medidas de rendimiento del método propuesto basado en detección de bordes con y sin la técnica de segmentación de brillo y el método de Canny propuesto en [34], utilizando los datos de la [Tabla 1](#)

Clasificación	Sen	Spec	Acc	Prec	FM
Sin segmentación de brillo	0.80	0.67	0.74	0.71	0.75
Con segmentación de brillo	0.80	0.71	0.75	0.73	0.76
Canny	0.96	0.71	0.83	0.76	0.85

Como se esperaba, y según los datos de la [Tabla 1](#), el uso de la técnica de segmentación de brillo reduce el número de fotogramas clasificados erróneamente como informativos. Además, como muestra la [Tabla 2](#), con la reducción de los falsos informativos, se incrementan los valores de especificidad y precisión.

4.3.4 Evaluación del Método de Análisis de Textura

El método de clasificación basado en análisis de textura, presentado en la [Sección 4.2](#), se evalúa bajo tres condiciones diferentes como lo ilustra la [Figura 31](#): La primera condición de evaluación (ver [Figura 31\(A\)](#)) establece una ventana del mismo tamaño de la imagen para calcular el vector de LBP, es decir, que este será de 256 posiciones. En la segunda evaluación (ver [Figura 31\(B\)](#)) se divide la imagen en 16 cuadrados iguales y por cada uno de ellos se calcula un descriptor LBP, por lo tanto el vector final tendrá 4096 posiciones. Finalmente

(ver [Figura 31\(C\)](#)), al igual que en la anterior condición, la imagen es dividida en 16 cuadrados pero esta vez solo se calcularán los descriptores de los 4 cuadrados centrales, dado que en ellos se concentra la mayor cantidad de información, lo que quiere decir que este vector tendría una longitud de 1024 posiciones.

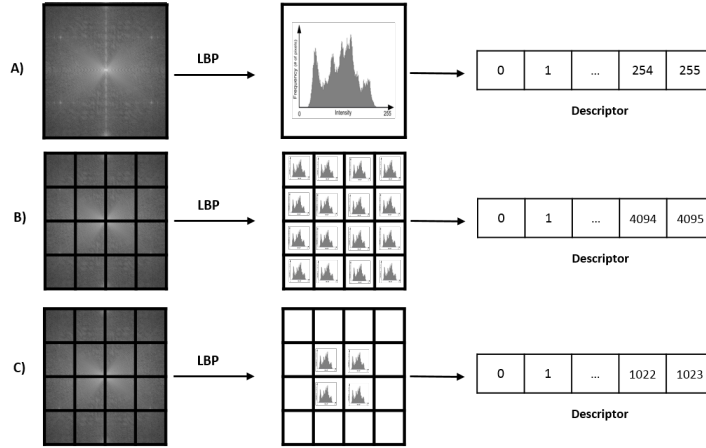


Figura 31: Condiciones de evaluación para la clasificación por análisis de textura

4.3.4.1 Resultados con *k-means*

En la [Tabla 3](#) se muestran los resultados para la clasificación de los fotogramas utilizando el algoritmo de particionamiento *k-means*. La evaluación se realiza en cada uno de los tres escenarios propuestos para el método basado en descriptores de textura: utilizando una ventana del mismo tamaño de la imagen, dividiendo la imagen en 16 bloques y por último tomando los 4 bloques centrales. En la [Tabla 4](#) se presentan los resultados de las medidas de rendimiento.

Tabla 3: Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes.

Clasificación	TI	FI	TNI	FNI
1 bloque (imagen completa)	270	64	236	30
16 bloques	293	183	117	7
4 bloques centrales	277	46	254	23

Tabla 4: Medidas de rendimiento del método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes., utilizando los datos de la [Tabla 3](#)

Clasificación	Sen	Spec	Acc	Prec	FM
1 bloque (imagen completa)	0.90	0.79	0.84	0.81	0.85
16 bloques	0.98	0.39	0.68	0.61	0.75
4 bloques centrales	0.92	0.85	0.88	0.86	0.89

Como se puede observar en los datos de la [Tabla 4](#), los mejores valores para las medidas se obtienen cuando el descriptor es calculado a partir de los 4 bloques centrales. En el segundo escenario de evaluación, k-means no realiza una buena clasificación porque se introduce mucho ruido en el descriptor y esto hace que los datos queden dispersos.

4.3.4.2 Resultados con SVM

La [Tabla 5](#) y la [Tabla 6](#) presentan los resultados respectivos para la predicción y clasificación de los fotogramas utilizando el algoritmo de aprendizaje supervisado SVM. La evaluación se realiza, tanto en predicción como en generalización, en cada uno de los tres escenarios propuestos para el método basado en descriptores de textura: con 1 bloque, es decir, utilizando una ventana del mismo tamaño de la imagen, dividiendo la imagen en 16 bloques y por último tomando los 4 bloques centrales. En la [Tabla 7](#) se presentan los resultados de las medidas de rendimiento para la predicción y en la [Tabla 8](#) los de la clasificación.

Para construir el modelo de clasificación se utilizaron 480 fotogramas (el 80 % del conjunto) y 120 (el 20 % restante) para las pruebas de predicción. La evaluación de la clasificación con SVM se realizó utilizando 200 nuevos fotogramas (100 informativos y 100 no informativos) de un video diferente al del conjunto inicial.

Tabla 5: (Predicción) Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes.

Clasificación	TI	FI	TNI	FNI
1 bloque (imagen completa)	47	5	55	13
16 bloques	40	7	53	20
4 bloques centrales	50	2	58	10

Tabla 6: (Generalización) Número de fotogramas, de los tres videos, clasificados por el método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes.

Clasificación	TI	FI	TNI	FNI
1 bloque (imagen completa)	89	8	92	11
16 bloques	97	62	38	3
4 bloques centrales	93	3	97	7

Tabla 7: (Predicción) Medidas de rendimiento del método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes., utilizando los datos de la [Tabla 5](#)

Clasificación	Sen	Spec	Acc	Prec	FM
1 bloque (imagen completa)	0.78	0.92	0.85	0.90	0.84
16 bloques	0.67	0.88	0.77	0.85	0.75
4 bloques centrales	0.83	0.97	0.90	0.96	0.89

Tabla 8: (Generalización) Medidas de rendimiento del método propuesto basado en descriptores de textura en tres escenarios de evaluación diferentes., utilizando los datos de la [Tabla 6](#)

Clasificación	Sen	Spec	Acc	Prec	FM
1 bloque (imagen completa)	0.89	0.92	0.90	0.92	0.90
16 bloques	0.97	0.38	0.67	0.61	0.75
4 bloques centrales	0.93	0.97	0.95	0.97	0.95

Las tablas de predicción ([Tabla 5](#) y [Tabla 7](#)) muestran el comportamiento del modelo con los datos restantes del mismo grupo del entrenamiento y las tablas de generalización ([Tabla 6](#) y [Tabla 8](#)) muestran qué tan eficiente es el modelo cuando clasifica datos nuevos. En términos generales se puede decir que la clasificación de fotogramas con [SVM](#) tiene buenos resultados, destacando la evaluación en el tercer escenario.

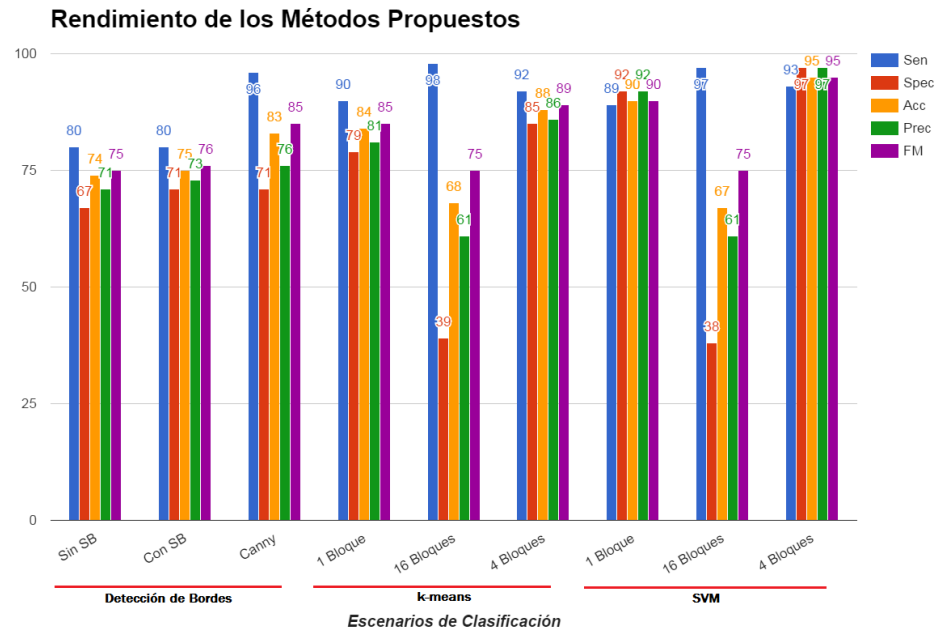


Figura 32: Rendimiento de los métodos de clasificación

4.3.5 Análisis y Comparación de los Métodos

La [Figura 32](#) presenta un gráfico comparativo de las medidas de rendimiento — para cada escenario — entre la [Tabla 2](#), la [Tabla 4](#) y la [Tabla 8](#). En él se puede observar que, para el caso de la clasificación con detección de bordes, se tienen mejores valores de sensibilidad y especificidad utilizando la técnica de segmentación de brillo. Además, los resultados del método de Canny — presentado en [34] — tienen un comportamiento similar a los del método presentado en este trabajo de grado, aunque con un poco de dispersión entre las medidas.

En cuanto a la clasificación no supervisada utilizando el algoritmo k-means se nota un aumento en la especificidad, precisión y exactitud en el escenario de los 4 bloques centrales. También, aunque la sensibilidad en el escenario de los 16 bloques está en el 96 %, los valores de las demás medidas son relativamente bajos, probablemente debido a que en este escenario se introduce mucho ruido al descriptor [LBP](#).

La clasificación supervisada con Support Vector Machine ([SVM](#)) presenta los mejores valores para todas las medidas en el escenario con 4 bloques. Sin embargo, cuando se clasifica dividiendo la imagen en 16 bloques se tienen valores muy dispersos al igual que en la clasificación con k-means, quiere decir que el descriptor [LBP](#) tomado de esta forma contiene información con mucho ruido que produce una mala clasificación.

4.3.6 Comparación entre Clasificación Supervisada y No Supervisada

Para el caso de la clasificación de fotogramas en videos de colonoscopia utilizando los algoritmos k-means y SVM, se pueden establecer ciertas ventajas tanto para la clasificación supervisada (SVM) como la no supervisada (k-means). Por ejemplo, la Figura 32 muestra que los mejores resultados se obtuvieron con SVM, sin embargo, este algoritmo depende fuertemente del modelo construido en el entrenamiento y, dado que los fotogramas de colonoscopia tienden a ser muy variables, es necesario actualizar constantemente el modelo con información nueva y diferente para conservar estos resultados. A diferencia de k-means, que divide el total de fotogramas en informativos y no informativos de manera automática obteniendo también buenos valores en las medidas de desempeño, la desventaja que presenta k-means es que puede llegar a ser no determinista, y elegir centroides diferentes en cada nueva ejecución, por lo que la consistencia de los resultados se vería comprometida, además un conjunto muy grande de fotogramas puede implicar un alto costo computacional y mucho tiempo en la clasificación.

En general, la decisión de utilizar un algoritmo supervisado o no supervisado depende de la cantidad de fotogramas que se necesiten procesar, es decir, si el conjunto es muy grande, sería una buena opción construir un buen modelo que los clasifique. En caso contrario, una separación automática con k-means podría ser una buena alternativa.

4.3.7 Elección del Mejor Método de Clasificación

Dado que, para este trabajo de grado, el propósito de la clasificación de fotogramas de colonoscopia es eliminar de los videos las imágenes no informativas, se determina el método que mejor cumple este objetivo en términos de la especificidad (ver Ecuación 19). Esta medida determina la correcta clasificación de los fotogramas no informativos, por lo tanto, un método con un alto valor de especificidad asegura que en el conjunto de los fotogramas informativos existe un número bajo de no informativos.

Con base en este criterio se puede asegurar que el mejor de los métodos es la clasificación basada en descriptores de textura con SVM (ver Subsección 4.2.4) y calculando el descriptor LBP de cada uno de los 4 cuadrados centrales de la imagen (ver Figura 31C). La especificidad de este método en clasificación con la tercera condición de evaluación (ver Tabla 8) es del 97 %, quiere decir que el modelo construido responde de manera eficiente a la clasificación de nuevas imágenes.

HERRAMIENTA DE ANOTACIÓN DE VIDEOS DE COLONOSCOPIA

En este capítulo se presenta el desarrollo de la aplicación de anotaciones de videos de colonoscopia. Inicialmente se describe la propuesta general, luego el proceso de selección de las herramientas que se utilizaron en el desarrollo, en tercer lugar se describe la implementación, por último se presentan la integración de la aplicación desarrollada con los métodos de clasificación y las pruebas de aceptación.

5.1 PROPUESTA GENERAL DE LA APLICACIÓN

En la actualidad se están entrenando algoritmos de inteligencia artificial para que reconozcan de manera automática diferentes lesiones, como cáncer o pólipos, en videos de colonoscopia [27],[28]. Estas técnicas construyen un modelo basado en fotogramas contenidos en anotaciones, que realiza un especialista sobre los videos. Una anotación equivale a una etiqueta realizada sobre un intervalo de tiempo en un video. En un video de colonoscopia, las etiquetas pueden ser de 3 tipos: *Cáncer*, *Pólipo* o *Normal*.

Con el fin de automatizar el proceso de anotación, se desarrolló una aplicación para gestionar los videos, anotaciones, pacientes y especialistas. El alcance de la aplicación se limita a que el proceso de anotación sea rápido y sencillo. A esta aplicación se le llamó Colonoscopy Annotation Tool (CAT).

5.2 SELECCIÓN DE HERRAMIENTAS DE DESARROLLO

La selección de las herramientas para implementar CAT fue un paso importante dentro del proceso de desarrollo, dado que la manipulación de videos suele convertirse en un desafío para la programación. Inicialmente se probaron algunos *frameworks* para aplicaciones de escritorio, como Processing [1], JMF [36] y JFMMPEG [23], que facilitarían la gestión de los videos y soportarían operaciones sobre ellos. Sin embargo, se encontró que estas herramientas estaban desactualizadas y presentaban fallas en tiempos de ejecución. Además, por ser productos para aplicaciones de escritorio, se pierden propiedades de software especiales como portabilidad y compatibilidad.

Por estas razones se exploraron herramientas para desarrollo web que soportarían manejo de archivos multimedia, apostando por la ventaja de las aplicaciones en la nube que no necesitan instalación para el usuario final. El resultado de la selección fue: el estándar más reciente



Figura 33: Integración de las tecnologías seleccionadas

te para visualización de páginas web HTML5, el lenguaje para páginas web Javascript y su *framework* JQuery para el control sobre los componentes de la aplicación. Se utilizaron elementos de Bootstrap para que la interfaz gráfica tuviera un diseño *responsive* y se adaptara a cualquier dispositivo, ya sean tablets, smartphones, computadores portátiles o de escritorio. Los videos se visualizan con la versión PRO del reproductor JW Player, el cual utiliza el protocolo RTMP para transmisión de videos. Se utilizó PostgreSQL como motor de base de datos y PHP con AJAX para la manipulación de las consultas SQL de manera asíncrona. Finalmente, la aplicación fue instalada sobre el servidor NGINX que, a diferencia de Apache, ofrece una implementación del protocolo de transmisión de videos RTMP, lo que facilita y mejora la reproducción de los videos de colonoscopia en la aplicación. La Figura 33 ilustra la integración de las tecnologías seleccionadas para desarrollar CAT.

5.3 IMPLEMENTACIÓN

Para el desarrollo de la aplicación se siguió la metodología ágil XP, por ser una de las que mejor se adapta a proyectos web. Con el fin de facilitar la gestión de las historias de usuario y el plan de entregas se utilizó la herramienta eXPert (cedesoftexpert.com). La implementación de CAT tuvo la valiosa colaboración del Ingeniero Heber Iván Mejía de la Universidad Señor de Sipan (Perú), quien realizó los primeros *mockups* de la aplicación (ver Figura 34), brindó asesoría en el uso de las diferentes tecnologías y desarrolló gran parte del módulo de gestión de usuarios y videos.

A continuación se presentan la arquitectura, los módulos que componen la aplicación, sus actores, las historias de usuario más relevantes, el release plan del proyecto y un detalle de la interfaz gráfica de CAT.

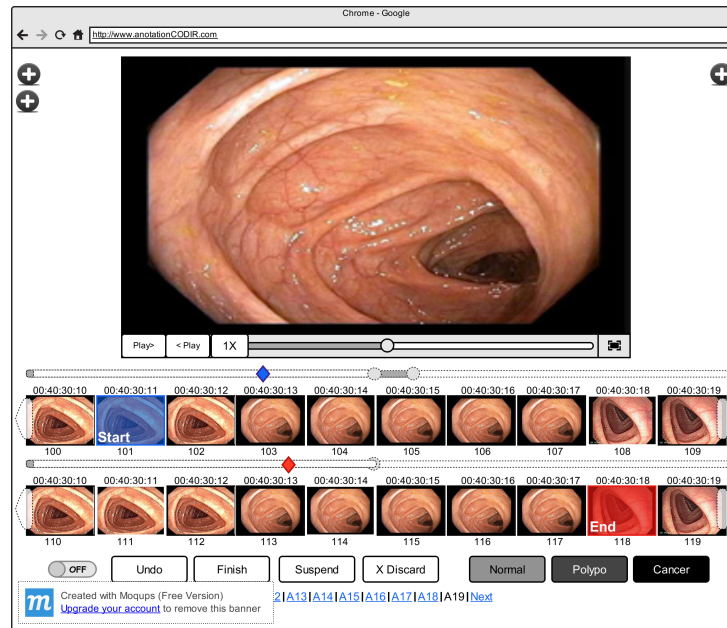


Figura 34: Primer diseño en *Mocqup* de la interfaz gráfica de la aplicación de anotaciones CAT

5.3.1 Actores

Los actores son las personas que actúan directamente con la aplicación. Para el caso de CAT se tienen los siguientes:

- **Administrador:** Es el encargado de administrar la base de datos y principalmente de asignar los videos a cada especialista.
- **Especialista:** Generalmente un médico gastroenterólogo que se encarga de realizar las anotaciones en los videos de colonoscopia.

5.3.2 Arquitectura

La Figura 35 ilustra la arquitectura general de CAT. El administrador es quien asigna, desde el programa, los videos para que el especialista pueda interactuar con la aplicación realizando anotaciones. El aplicativo debe realizar operaciones de consulta y manejo de componentes en el sistema de archivos para luego mostrar resultados al usuario final.

5.3.3 Módulos

CAT se divide en 7 módulos, especializados en diferentes funciones con el fin de tener una mejor organización y mayor control sobre el desarrollo de la aplicación. A continuación se listan y describen cada uno de ellos:

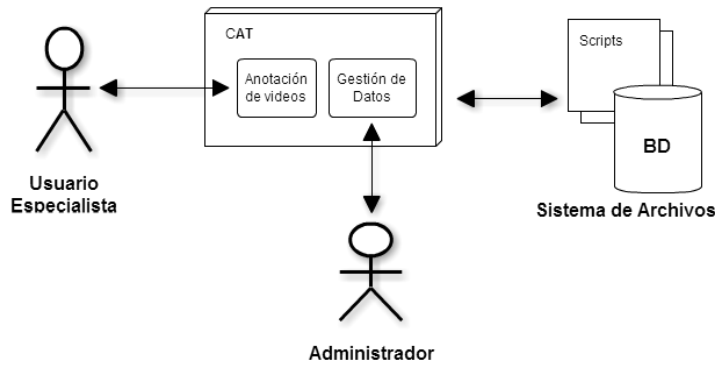


Figura 35: Arquitectura general de CAT

- **GUI:** Es el módulo donde se implementan las vistas y se controlan los componentes de la interfaz gráfica de la aplicación.
- **Gestión de Usuarios:** Este módulo contiene las funcionalidades **CRUD** para administrar todos los usuarios/actores de la aplicación.
- **Gestión de Videos:** Este módulo contiene las funcionalidades **CRUD** para administrar los videos, incluyendo la de asignación a especialistas.
- **Gestión de Anotaciones:** Este módulo es el encargado de administrar con operaciones **CRUD** las anotaciones que los especialistas realicen sobre los videos de colonoscopia.
- **Controles de Reproducción:** En este módulo se encuentran implementados los controles de reproducción del reproductor de video, además de los básicos se han creado algunos personalizados.
- **Gestión de Fotogramas:** Este módulo es el encargado de la visualización y control de los *thumbnails* del video en una línea de tiempo.
- **Reportes:** Este módulo se encarga de entregar el informe de anotaciones por video.

5.3.4 Historias de Usuario

La tabla 9 muestra el listado de las historias de usuario más relevantes de la aplicación.

Tabla 9: Listado de historias de usuario

Identificador	Nombre	Descripción
HU01	Crear un usuario	Como administrador quiero crear un usuario para que pueda hacer uso del sistema
HU02	Modificar un usuario	Como administrador quiero modificar los datos de un usuario para actualizar su información
HU03	Borrar un usuario	Como administrador quiero borrar un usuario para que no pueda hacer más uso del sistema
HU04	Realizar una anotación	Como especialista quiero realizar una anotación en un segmento de un video para etiquetar una anomalía
HU05	Modificar una anotación	Como especialista quiero modificar una anotación para cambiar el segmento de video o el tipo de etiqueta
HU06	Deshacer una anotación	Como especialista quiero deshacer una anotación luego de que la haya creado para eliminarla del sistema
HU07	Subir un video	Como administrador del sistema quiero subir un video a la aplicación para que pueda ser utilizado

HUo8	Descartar fotogramas no informativos	Como administrador del sistema quiero eliminar los fotogramas no informativos de un video de colonoscopia para que pueda ser anotado por los especialistas
HUo9	Asignar videos	Como administrador quiero asignar un video a un especialista para que este pueda realizar las anotaciones
HU10	Reporte de anotaciones	Como administrador del sistema quiero generar un reporte por asignación para obtener un listado de las anotaciones)

5.3.5 Plan de Entregas - “Release Plan”

A continuación, en la [Tabla 10](#), se presenta el plan de entregas donde se establece la prioridad de implementación de las historias de usuario planteadas en la [Tabla 9](#). El desarrollo se realiza en 4 grandes iteraciones que son organizadas de acuerdo a la importancia de las historias de usuario.

Tabla 10: Plan de entregas para el desarrollo de la aplicación [CAT](#)

Iteración	Historias de Usuario	Objetivo
1	▪ HU01 ▪ HU02 ▪ HU03	Implementar el CRUD de usuarios del sistema

2	■ HU07 ■ HU09	Implementar las funciones de carga y asignación de videos
3	■ HU04 ■ HU05 ■ HU06	Implementar el módulo de anotaciones
4	■ HU08 ■ HU010	Acoplar los métodos de eliminación de fotogramas con la aplicación y generar el reporte de anotaciones

5.3.6 Interfaz Gráfica

En esta sección se muestran las interfaces gráficas más importantes de la aplicación de anotaciones.

El acceso a la aplicación se realiza desde la interfaz mostrada en la [Figura 36](#), la cual se compone principalmente de un campo para ingresar el usuario y otro para la contraseña. El sistema valida internamente, mediante una consulta SQL ejecutada con PHP, si el usuario corresponde a un administrador o un especialista y dependiendo del caso redirecciona la aplicación a la vista correspondiente.

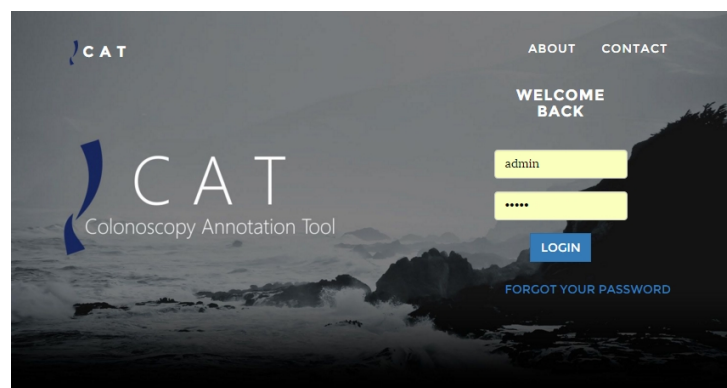


Figura 36: Ventana de ingreso a la aplicación

La ventana general de la aplicación para anotaciones es presentada en la [Figura 37](#). En esta primera vista se muestra un menú desplegable, en la parte izquierda de la pantalla, donde se encuentran los vi-

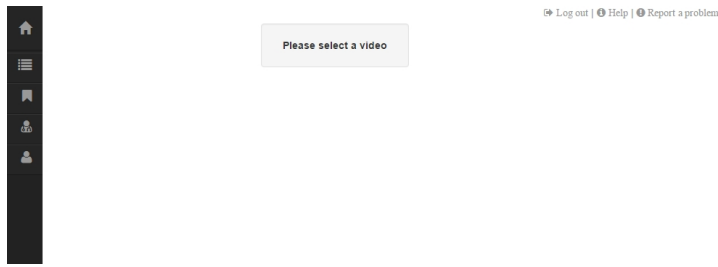


Figura 37: Ventana general de la aplicación para anotaciones

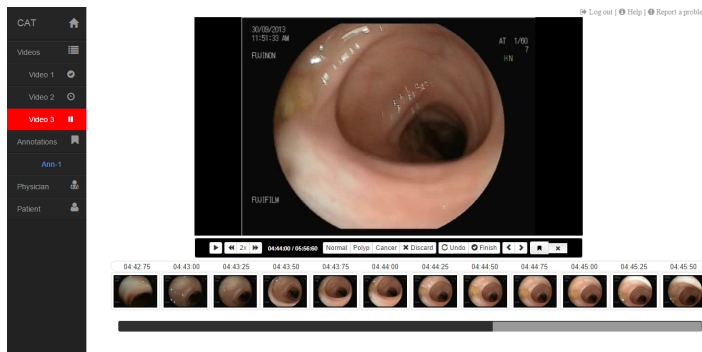


Figura 38: Ventana de la aplicación cuando se ha seleccionado un video

deos asignados al especialista, una lista de las anotaciones realizadas por video, información para el especialista sobre los videos pendientes, pausados o finalizados y los datos más relevantes del paciente al que se le realizó el procedimiento.

Una vez que el especialista selecciona uno de los videos de la lista que le ha sido asignada por un administrador, se cargan en la vista: el reproductor, los controles de anotación, reproducción, la barra de progreso del video y una fila de *thumbnails* o línea de tiempo donde se selecciona el intervalo de anotación y se previsualiza el video (ver Figura 38).

Con el fin de tener un manejo personalizado de la reproducción del video y de la línea de tiempo, se implementaron controles y algunos botones con funcionalidades específicas. De izquierda a derecha en la Figura 39 está el botón básico de *play/pause*, los botones que controlan la velocidad, hacia adelante o hacia atrás, de reproducción, una etiqueta que muestra el tiempo actual y el total del video, 4 botones para indicar el tipo de anotación seleccionada en la línea de tiempo: *Normal*, *Polyp*, *Cancer* o *Discard* en caso de tratarse de fotografías que deban ser descartados. Luego se encuentra un botón para deshacer la última anotación realizada y otro para finalizar el proceso de anotación con el video actual. Siguen dos botones que avanzan un *thumbnail* a la izquierda o a la derecha de la línea de tiempo. Por último se implementó un botón tipo *switch* para indicar si se va a realizar una anotación sobre la línea de tiempo (ON) o si se va a ver en la pantalla del reproductor uno de las *thumbnails* (OFF).



Figura 39: Detalle de la barra de control de reproducción

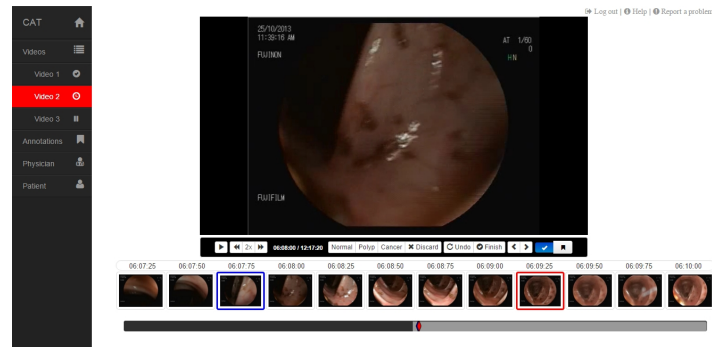


Figura 40: Selección de un intervalo de anotación

Una vez que el especialista haya elegido el video, y si este no ha sido finalizado, se puede realizar una anotación dando clic en el *thumbnail* de la línea de tiempo donde se considere que deba iniciar el intervalo de tiempo de la anotación, este inicio se marcará con un cuadro de color azul al rededor de la imagen que se seleccione. De manera similar el final del intervalo se marca con color rojo. Un ejemplo de anotación se muestra en la [Figura 40](#).

Cualquier anotación puede ser modificada en sus datos como el tiempo inicio, el tiempo final del intervalo o el tipo de anomalía etiquetada. Para editar una anotación es necesario seleccionarla de la lista del panel izquierdo y realizar la respectiva modificación volviendo a hacer el proceso explicado anteriormente. La [Figura 41](#) corresponde a una modificación de la anotación de la [Figura 40](#), donde se ha cambiado el final del intervalo de tiempo.

La [Figura 42](#) presenta la vista general del panel de administración, en donde se pueden realizar las tareas de gestión principalmente de usuarios y videos.

En el panel izquierdo de la vista de administrador se puede seleccionar la entidad que se quiere gestionar. En el caso de agregar un

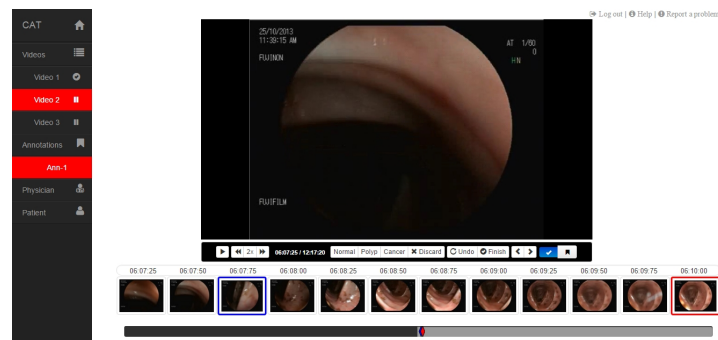


Figura 41: Vista de la aplicación en el evento de editar una anotación

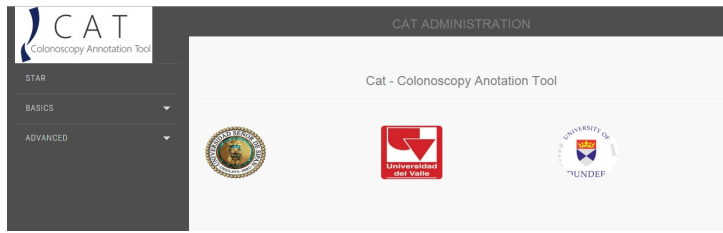


Figura 42: Ventana principal del administrador

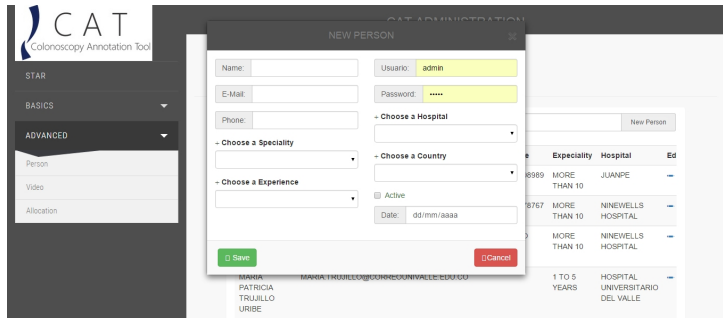


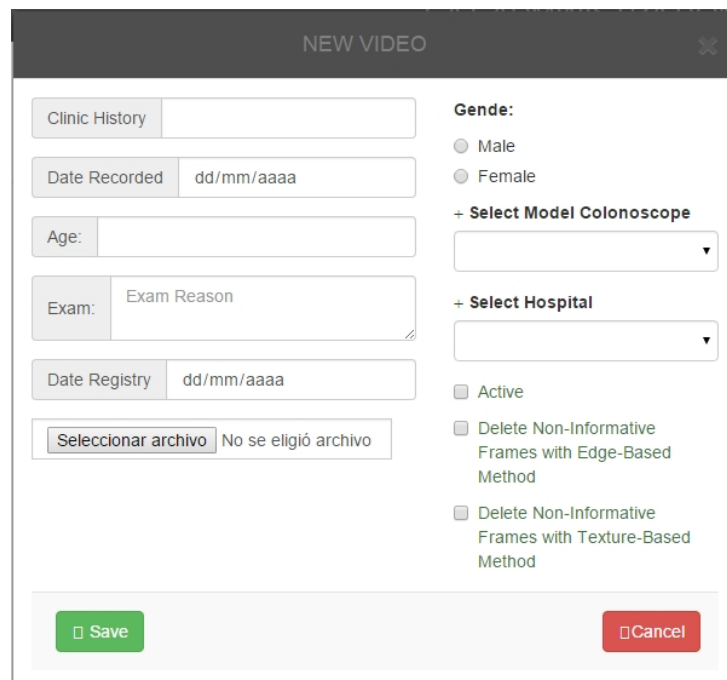
Figura 43: Cuadro de diálogo para el evento de agregar un nuevo usuario

nuevo usuario, se despliega un cuadro de diálogo como el de la [Figura 43](#) donde se piden, a parte de los datos básicos, un nombre de usuario, contraseña y se debe seleccionar el tipo de actor del sistema: especialista o administrador.

Si se desea agregar un nuevo video a la aplicación, en el panel izquierdo se encuentra la opción que despliega un cuadro de diálogo como el de la [Figura 44](#). Los datos requeridos para el nuevo video son, entre otros, el número de historia clínica, la fecha de grabación, edad del paciente y la razón por la que se realizó el procedimiento. Llenados estos campos el usuario selecciona el archivo de video y en la parte derecha debe indicar el tipo de algoritmo con el que serán eliminados los fotogramas no informativos del video.

El requerimiento de la asignación de videos a especialistas por parte de un administrador se desarrolló con la vista que se muestra en la [Figura 45](#), en donde se puede elegir de una lista de especialistas el usuario al que se le quiera asignar uno o varios videos. La asignación se realiza en el panel inferior que aparece cuando es seleccionado el especialista. Una vez asignado el video, la etiqueta *unallocated* cambia por *allocated* tal como se muestra en la [Figura 46](#).

Finalmente, cuando el especialista realice por lo menos una anotación en el video que se le ha asignado, el administrador podrá descargar un reporte, en formato de texto **txt**, con las anotaciones que se hayan hecho. La forma de descargar este reporte es por medio de un enlace (*annotations*) que aparece en el cuadro de la asignación del video al especialista, tal como lo muestra la [Figura 47](#).



NEW VIDEO

Clinic History

Date Recorded

Age:

Exam:

Date Registry

No se eligió archivo

Gende:

☐ Male

☐ Female

+ Select Model Colonoscope

+ Select Hospital

☐ Active

☐ Delete Non-Informative Frames with Edge-Based Method

☐ Delete Non-Informative Frames with Texture-Based Method

Figura 44: Cuadro de diálogo para el evento de agregar un nuevo video

Cat - Allocation

Buscar

Name	Estate	Asignar
ALBERTO		
FIGO		
CRISTIAN JAVIER BALLESTEROS		
MARIA PATRICIA TRUJILLO URIBE		
HEBER IVAN MEJIA CABRERA		

ALBERTO

History Clinic	Link	Video State	Completed	Action
HUV-1-01-13	HTTP://127.0.0.1/CAT/VIDEOS/VIDEO1.MP4	UNALLOCATED	UNALLOCATED	
HUV-1-01-13	HTTP://127.0.0.1/CAT/VIDEOS/VIDEO1.MP4	UNALLOCATED	UNALLOCATED	
HUV-1-01-13	HTTP://127.0.0.1/CAT/VIDEOS/VIDEO1.MP4	UNALLOCATED	UNALLOCATED	

Figura 45: Ventana de la aplicación para el evento de asignar un video a un especialista

ALBERTO

History Clinic	Link	Video State	Completed	Action
HUV-1-01-13	HTTP://127.0.0.1/CAT/VIDEOS/VIDEO1.MP4	ALLOCATED	UNALLOCATED	

Figura 46: Detalle de la asignación de un video a un especialista

ALBERTO

History Clinic	Link	Video State	Completed	Action
HUV-1-01-13	HTTP://127.0.0.1/CAT/VIDEOS/VIDEO1.MP4	ALLOCATED	UNALLOCATED	Annotations

Figura 47: Enlace *annotations* para descargar el reporte de anotación

5.4 INTEGRACIÓN DE LA APLICACIÓN CON LOS MÉTODOS DE CLASIFICACIÓN DE FOTOGRAMAS

Con el fin de entregarle al especialista una versión del video con la menor cantidad de fotogramas no informativos posible, se realiza una integración de la aplicación de anotaciones [CAT](#), con los métodos de clasificación de fotogramas propuestos en el [Capítulo 4](#).

Tal como aparece en la [Figura 44](#), el administrador puede elegir entre dos de los métodos de clasificación propuestos: Basada en detección de bordes con segmentación de brillo ([Sección 4.1](#)) y basada en descriptores de textura con k-means ([Sección 4.2](#)). Se decidió acoplar estos dos métodos dado que no necesitan, como en el caso de [SVM](#), de un entrenamiento para clasificar.

Para que la integración de los métodos de clasificación con [CAT](#) sea posible, se debe instalar en el servidor de la aplicación el software, librerías y demás componentes que requieran los métodos mencionados. En el caso del método por bordes, que fue implementado en QT, C++ y OpenCV, es necesario instalar y configurar los paquetes de QT5, MinGW y OpenCV2,4 o superior. También es necesario instalar Octave4,0 o superior con el paquete de procesamiento de imágenes: *image*, dado que el método basado en texturas fue implementado como una función de ese lenguaje. Además se requiere que esté instalada la librería FFMPEG para descomponer los videos en todos sus fotogramas y luego hacer uno nuevo solo con los informativos.

La ejecución de los métodos se hace mediante línea de comandos utilizando la función `exec()` de PHP.

5.5 PRUEBAS DE ACEPTACIÓN

Las pruebas realizadas al sistema fueron principalmente de caja negra, es decir, se aplicaron principalmente a las funcionalidades gráficas de la aplicación que a la parte lógica, dado que los algoritmos de utiliza [CAT](#) fueron evaluados por separado en la [Sección 4.3](#) del [Capítulo 4](#).

Las pruebas de aceptación que se aplicaron a cada historia de usuario se describen a continuación en la [Tabla 11](#).

Tabla 11: Pruebas de aceptación de la aplicación CAT

Cód. Prueba	Historia Usuario	Acción	Resultado Esperado	Resultado Obtenido
Po1	HUo1	Llenar los campos del nuevo usuario y presionar el botón de <i>crear</i>	Nuevo registro en la base de datos	Datos del nuevo usuario almacenados correctamente como un nuevo registro en la base de datos
Po2	HUo2	Cambiar un dato de un usuario y presionar el botón de <i>modificar</i>	Registro del usuario actualizado en la base de datos	El dato fue modificado en la base de datos exitosamente
Po3	HUo3	Presionar el botón de <i>borrar</i> de un usuario	El registro del usuario es eliminado de la base de datos	El usuario ya no aparece en la base de datos
Po4	HUo4	Seleccionar un intervalo de tiempo en la fila de <i>thumbnails</i> y presionar el botón de la etiqueta correspondiente a la anotación	Intervalo de tiempo almacenado en la base de datos	Se creó un nuevo registro en la tabla de anotaciones que guarda los datos de la anotación
Po5	HUo5	Seleccionar la anotación, modificar sus datos y presionar el botón de la etiqueta correspondiente	El registro de la anotación es modificado con los nuevos datos	Los datos de la anotación fueron modificados exitosamente
Po6	HUo6	Presionar el botón <i>deshacer</i>	La última anotación es eliminada de la base de datos	La última anotación realizada por el especialista ya no aparece en la base de datos

Po7	HUo7	Seleccionar un archivo de video, llenar los datos y presionar el botón <i>crear video</i>	El video se almacena en el servidor y se crea un registro en la base de datos con la información del video	El video aparece en la carpeta de videos en el servidor y el registro con la información del video es creado exitosamente
Po8A	HUo8	Seleccionar el método de descarte de fotogramas por detección de bordes	Se almacena en el servidor un nuevo video creado con los fotogramas informativos arrojados por el método de bordes	El nuevo video es almacenado exitosamente
Po8B	HUo8	Seleccionar el método de descarte de fotogramas por descriptores de textura con k-means	Se almacena en el servidor un nuevo video creado con los fotogramas informativos arrojados por el método de textura	El nuevo video es almacenado exitosamente
Po9	HUo9	Seleccionar un especialista y presionar el botón de <i>asignar</i> en uno de los videos	Se crea un nuevo registro en la tabla de asignaciones que relaciona al especialista con el video	El registro en la tabla de asignaciones es creado exitosamente
P10	HU10	Seleccionar la asignación de un video a un especialista y presionar el botón de reportes	Listado de las anotaciones realizadas al video por parte del especialista	El listado se generó correctamente

En este trabajo de grado se abordaron los temas de clasificación de fotogramas y anotación de videos de colonoscopia, por lo que se implementaron métodos que separan las imágenes en informativas y no informativas y se desarrolló una aplicación que facilita el proceso de anotación de videos de colonoscopia. A continuación se presentan las respectivas conclusiones de cada tema.

6.1 CONCLUSIONES SOBRE LOS MÉTODOS DE CLASIFICACIÓN

Como solución al problema planteado se proponen principalmente dos métodos: el primero bajo la premisa que un fotograma informativo, dado que tiene mayor contenido, tienen un mayor número de bordes que uno no informativo. El segundo método toma en cuenta el dominio de las frecuencias de los fotogramas y extrae características de textura para clasificarlos con técnicas de inteligencia artificial.

El método propuesto basado en detección de bordes utiliza el detector Sobel y define la medida **EPP** para calcular el porcentaje de píxeles de borde presentes en los fotogramas. Con esta medida y un umbral especificado previamente, se determina el tipo de fotograma: *informativo* si el **EPP** es mayor que el umbral o *no informativo* en caso contrario. Este método no clasifica todos los fotogramas de los videos en forma correcta, debido a los falsos bordes producidos por la reflexión de la luz del colonoscopio con el agua utilizada en el procedimiento, por lo cual se propone una técnica de detección de falsos fotogramas informativos. La detección de los *falsos informativos* se realiza con una técnica de segmentación de brillo, que consiste en segmentar las zonas de la imagen con una alta intensidad en niveles de gris y extraer sus bordes. La detección de falsos informativos utiliza el porcentaje de los bordes con la medida **PBP** y se resta del **EPP** inicial. Finalmente esta diferencia es comparada con el umbral definido al inicio y se determina, de la misma manera anterior, si el fotograma es informativo o no. En la evaluación del método se compara el uso y no uso de la técnica de segmentación de brillo, dando mejores resultados la clasificación con un postprocesamiento basado en brillo.

El segundo método de clasificación utiliza el descriptor **LBP**, sobre la imagen de la transformada de Fourier de cada fotograma, para extraer información de textura y emplearla como característica en los algoritmos de aprendizaje supervisado y no supervisado: **SVM** y *k-means*. Para ambos clasificadores se toman tres escenarios distintos de evaluación: Primero se calcula el **LBP** con una ventana de la misma

dimensión que la imagen, el segundo escenario corresponde a una división de la imagen en 16 bloques cuadrados para los cuales se calcula un vector de LBP por cada uno y se junta en uno solo al final. El tercer y último escenario es similar al anterior, solo que se calcula un vector de características para la imagen solo con los 4 cuadrados centrales.

Los resultados muestran que, de todas las técnicas, tiene mayor precisión (97 %) la clasificación con SVM en el tercer escenario de evaluación. Finalmente, vale la pena señalar que la calidad de resolución del video capturado en el procedimiento de colonoscopia influye, de gran manera, en la clasificación de los fotogramas. Dado que la baja calidad puede ser vista como un tipo de borrosidad y, en el caso del método basado en detección de bordes, elimina información importante de los fotogramas causando una mala clasificación.

En síntesis, con base en el desarrollo y análisis de los resultados de los métodos de clasificación de fotogramas de colonoscopia, se puede afirmar que:

- La clasificación de fotogramas de videos de colonoscopia, reduce el número de fotogramas que no aportan información al diagnóstico de anomalías en el colon, lo que aumenta los valores de precisión en algoritmos de detección automática de lesiones, tales como, cáncer o pólipos.
- Con la eliminación de los fotogramas informativos de los videos, se reduce el tiempo que un especialista debe invertir para etiquetar anomalías que pueda encontrar.
- De la misma manera, eliminar los fotogramas no informativos de los videos reduce el tamaño de los mismos y se ahorra almacenamiento en disco.
- La clasificación por detección de bordes es un método que presenta buenos resultados y es de los más fáciles de implementar.
- Utilizar una técnica de segmentación/eliminación de brillo reduce el número de fotogramas no informativos clasificados incorrectamente como informativos.
- La clasificación con k-means y SVM tienen mejores resultados cuando se forma el descriptor LBP con los 4 bloques centrales, esto se debe a que la información relevante se concentra en el centro de la imagen.
- Formar el descriptor LBP dividiendo el fotograma en 16 bloques produce una mala clasificación, probablemente porque de esta forma se introduce mucho ruido en el descriptor.
- La elección del mejor método se realizó con base en los valores de especificidad, dado que se requiere eliminar la mayor

cantidad de fotogramas no informativos de los videos y esta medida determina el porcentaje de estos fotogramas que son clasificados correctamente. El método con el mejor valor de especificidad fue la clasificación con [SVM](#) utilizando los 4 bloques centrales.

6.2 CONCLUSIONES SOBRE LA APLICACIÓN DE ANOTACIONES

La aplicación de anotaciones ([CAT](#)) se desarrolló en colaboración con las Universidades de Dundee, en Escocia, y Señor de Sipan, en Perú, con el fin de generar un formato específico que sirva para obtener datos que contribuyan a la construcción de un modelo de detección automática de lesiones en el colon.

En la aplicación se identificaron dos actores: administrador y especialista. El primero es el encargado de la gestión de usuarios y asignación de videos, y el segundo es el médico especialista que realiza las anotaciones en el video. El proceso de anotación se realiza de manera sencilla seleccionando, en una línea de tiempo con *thumbnails*, el inicio y fin del intervalo donde se encuentra la anomalía. Un especialista puede anotar tantos videos como se le sean asignados y al final un administrador puede generar el reporte de anotación para ver la información pertinente que se requiere para la construcción del modelo de los algoritmos de detección automática. Además, con el fin de reducir el tiempo requerido para realizar las anotaciones, se integraron en [CAT](#) dos de los métodos de clasificación de fotogramas. Con esto, un administrador puede elegir uno de ellos y generar un nuevo video a partir del conjunto de fotogramas informativos que arrojó el método.

Las ventajas de portabilidad, disponibilidad, eficiencia y nulo proceso de instalación que ofrecen las aplicaciones web, fueron determinantes para elegir el desarrollo en la nube antes que el de escritorio. La aplicación de anotación de videos de colonoscopia se implementó usando las más recientes tecnologías web que soportan contenido multimedia, tales como: HTML5, Javascript, PHP, entre otras.

Finalmente se debe anotar que para el desarrollo de aplicaciones de uso médico es necesario tener presente que la usabilidad debe ser una prioridad, dado que estos usuarios requieren agilidad y practicidad en los procesos de su trabajo.

TRABAJO FUTURO

La clasificación de fotogramas en videos de colonoscopia sigue siendo un problema de investigación abierto, en donde se busca desarrollar métodos lo suficientemente robustos como para clasificar correctamente las imágenes de cualquier video. Además, en la actualidad existen muchos clasificadores, descriptores y extractores de características —y aparecerán cada día más— con los cuales se puedan mejorar los resultados obtenidos en este trabajo de grado y aportar al mejoramiento de la calidad de vida humana.

En cuanto a la aplicación de anotaciones [CAT](#), además de integrar todos los métodos de clasificación de fotogramas que se desarrollen, se piensa acoplarla con algoritmos de detección automática de lesiones en el colon y convertirla en un asistente inteligente que sea capaz de encontrar en el video, de manera automática, lesiones o anomalías que quizás el especialista haya pasado por alto.

BIBLIOGRAFÍA

- [1] Processing 2. Processing, 2015. [Web; accessed on 27-12-2015 to www.processing.org].
- [2] Tinku Acharya and Ajoy K. Ray. *Image Processing: Principles and Applications*. WILEY INTERSCIENCE, Reading, New Jersey, 2005.
- [3] Adobe. Real-time messaging protocol (rtmp) specification, 2015. [Web; accessed on 06-11-2015 to www.adobe.com/devnet/rtmp.html].
- [4] ArcSoft. What is color space, 2015. [Web; accessed on 27-12-2015 to www.arcsoft.com/topics/photostudio-darkroom/what-is-color-space.html].
- [5] Olivier Aubert and Yannick Prié. Advène: active reading through hypervideo. [Web; accessed on 27-12-2015 to liris.cnrs.fr/advène/download.html].
- [6] Alberto Farca Belsaguy. ¿qué es un colonoscopio?, 2015. [Web; accessed on 09-30-2015 to farca.org/colonoscopia/colonoscopio.htm].
- [7] BTP. Bootstrap tutorial, 2015. [Web; accessed on 27-12-2015 to www.besttutorialpoint.com/bootstrap-tutorial].
- [8] M. Trujillo C. Ballesteros and C. Mazo. Automatic classification of non-informative frames in colonoscopy videos. *6th Latin American Conference on Networked and Electronic Media. LACNEM*, 2015. Facultad de Minas, Universidad Nacional sede Medellín.
- [9] Enrique Carmona. Tutorial sobre máquinas de vectores soporte. [Web; accessed on 27-12-2015 to [www.ia.uned.es/~ejcarmona/publicaciones/\[2013-Carmona\]%20SVM.pdf](http://www.ia.uned.es/~ejcarmona/publicaciones/[2013-Carmona]%20SVM.pdf)].
- [10] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [11] Deisy Chaves. Preprocesamiento de imágenes digitales de carbonizados, 2008. Trabajo de Grado.
- [12] G.P. Crawford. *Liquid Crystals: Frontiers in Biomedical Applications*.

- [13] EISC. Universidad del Valle. Programación extrema, 2015. [Web; accessed on 27-12-2015 to eisc.univalle.edu.co/materias/WWW/material/lecturas/xp.pdf].
- [14] Liga Contra el Cancer. Cancer colorrectal, 2015. [Web; accessed on 06-11-2015 to www.ligacontraelcancer.com.co/cancer-colorrectal/magnitud-del-problema].
- [15] FFMPEG. Ffmpeg, multimedia framework, 2015. [Web; accessed on 06-11-2015 to www.ffmpeg.org].
- [16] Jesse James Garrett. Ajax (programming), 2015. [Web; accessed on 27-12-2015 to [en.wikipedia.org/wiki/Ajax_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming))].
- [17] PostgreSQL Global Development Group. Postgresql, 2015. [Web; accessed on 06-11-2015 to www.postgresql.org].
- [18] The PHP Group. Php, 2015. [Web; accessed on 27-12-2015 to php.net/manual/es/intro-what-is.php].
- [19] Shanmugam K. Dinstein I. Haralick, R. Textural features for image classification. *ieee transactions on systems. Man and Cybernetics*, pages 610 – 621, 1973.
- [20] Marko Heikkil and Timo Ahonen. A general local binary pattern (lbp) implementation for matlab, 2014. [Web; accessed on 27-12-2015 to <http://www.cse.oulu.fi/CMV/Downloads/LBPMatlab>].
- [21] WCRF International. Colorectal cancer statistics, 2012. [Web; accessed on 06-11-2015 to www.wcrf.org/int/cancer-facts-figures/data-specific-cancers/colorectal-cancer-statistics].
- [22] Bernd Jahne. *Digital Image Processing, 5th Edition*. 2002.
- [23] JFFMPEG. Jffmpeg - ffmpeg for java, 2015. [Web; accessed on 27-12-2015 to jffmpeg.sourceforge.net].
- [24] jQuery Staff. About jquery, 2015. [Web; accessed on 27-12-2015 to jquery.com].
- [25] Venkata Praveen Karri and B. Tech. Effective and accelerated informative frame filtering in colonoscopy videos using graphic processing units. *M.S. thesis, University of North Texas*, 2010.
- [26] Dorsey Lopes. *Importan Concepts in Signal Processing, Image Processing and Data Compression*. University Publications, 2012. ISBN 978-81-323-3604-4.
- [27] S. Manivannan, R. Wang, E. Trucco, and A. Hood. Automatic normal-abnormal video frame classification for colonoscopy. *2013 IEEE 10th International Symposium on Biomedical Imaging (IS-BI)*, pages 644 – 647, 2013.

- [28] S. Manivannan, R. Wang, M. P. Trujillo, J. A. Hoyos, and E. Trucco. Video-specific svms for colonoscopy image classification. *Lecture Notes in Computer Science*, 8899:11 – 21, 2014.
- [29] Matlab. Create ggray level co-occurrence matrix, 2015. [Web; accessed on 27-12-2015 to www.mathworks.com/help/images/ref/graycomatrix.html].
- [30] Vclav Hlavac y Roger Boyle Milan Sonka. *Image processing, analysis and machine vision. Segunda Edición*.
- [31] Mozilla. About javascript, 2015. [Web; accessed on 27-12-2015 to developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript].
- [32] M. Neteler and H. Mitasova. *Open Source GIS: A Grass Gis Approach*.
- [33] Staff of what-when-how.com. Morphology, 2015. [Web; accessed on 27-12-2015 to what-when-how.com/introduction-to-video-and-image-processing/morphology-introduction-to-video-and-image-processing-part-1].
- [34] J. Oh, S. Hwang, J. Lee, W. Tavanapong, J. Wong, and Piet C. de Groen. Informative frame classification for endoscopy video. *Medical Image Analysis*, 11(1):110–127, 2007.
- [35] openclassroom. Svm linear classification. [Web; accessed on 27-12-2015 to openclassroom.stanford.edu/MainFolder/DocumentPage.php?course=MachineLearning&doc=exercises/ex7/ex7.html].
- [36] Oracle. Jmf - java media framework, 2015. [Web; accessed on 27-12-2015 to www.oracle.com/technetwork/java/javase/download-142937.htm].
- [37] N. Otsu. A threshold selection method from gray level histograms. *IEEE Trans. Systems, Man and Cybernetics*, 9:62–66, March 1979. minimize inter class variance.
- [38] Z. Guoying A. Timo P. Matti, H. Abdenour. *Computer Vision Using Local Binary Patterns*. Springer, London Dordrecht, 2011.
- [39] JW Player. The world’s most popular video player, 2015. [Web; accessed on 27-12-2015 to www.jwplayer.com].
- [40] D. M. W. Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 02(1):37 – 63, 2011.

- [41] N. Rungseekajee, M. Lohvithee, and I. Nilkhamhang. Informative frame classification method for real-time analysis of colonoscopy video. *ECTI-CON 2009. 6th International Conference*, 02(1): 1076 – 1079, 2009.
- [42] Igor Sysoev. Nginx server, 2015. [Web; accessed on 06-11-2015 to nginx.org].
- [43] Annotation Academic Video Team. Annotation academic video, . [Web; accessed on 27-12-2015 to entwinemedia.github.io/annotations].
- [44] OVA Team. Open video annotation, . [Web; accessed on 27-12-2015 to www.openvideoannotation.org].
- [45] VIA Team. Via: Video image annotation, . [Web; accessed on 27-12-2015 to sourceforge.net/p/via-tool/wiki/Home].
- [46] Cancer Research UK. Anal cancer statistics, 2012. [Web; accessed on 06-11-2015 to www.cancerresearchuk.org/health-professional/cancer-statistics/statistics-by-cancer-type/anal-cancer].
- [47] W. and M. Burger. *Principles of Digital Image Processing*. Springer, London Verlag, 2009.
- [48] w3.org. Html5, 2014. [Web; accessed on 27-12-2015 to www.w3.org/TR/html5].
- [49] Baopu Li Yichen Fan, Max Q.-H. Meng. A novel method for informative frame selection in wireless capsule endoscopy video. *33rd Annual International Conference of the IEEE EMBS*, 2011.