

Capítulo 1. Introducción

En la actualidad existen múltiples aplicaciones que utilizan el almacenamiento de imágenes en video, un ejemplo común son las cámaras de teléfonos inteligentes que tienen el hardware y software suficiente para grabar información en video, otro ejemplo son los sistemas de reconocimiento facial que con base en PCA (Principal Component Analysis) y LDA (Linear Discriminant Analysis) son capaces de reconocer e identificar un rostro en una imagen (Sinsinwar & Dwivedi, 2014). En el área de medicina (Chen, Meng, & Lin, 2013), mencionan la importancia de utilizar sistemas multimedia inteligentes que permitan recuperar video basado en un sistema web inteligente. Como podemos observar el campo de acción para estas aplicaciones es muy amplio, incluso puede combinar ramas de investigación: realidad virtual y medicina, donde utilizando simuladores virtuales se pueden realizar cirugías para entrenar a los futuros cirujanos (De Moraes & dos Santos Machado, 2007).

Sin embargo, existe un caso particular que es importante comentar ya que es un ejemplo muy claro de la necesidad de almacenamiento de la información que se captura; las cámaras de video vigilancia tradicionales capturan cada imagen que pasa a través del lente y la graban en dispositivos de almacenamiento los cuales tienen un límite en la cantidad de información que guardan. Este funcionamiento genera una cantidad masiva de información que puede ser catalogada como no relevante, gran parte de esta información son imágenes que no incluyen personas, las cuales son fundamentales en el contexto de los sistemas de seguridad, la presente propuesta plantea utilizar modelos de detección e identificación facial que permitan tomar decisiones al momento de almacenar información.

1.1 Descripción del problema

Existen múltiples dispositivos de almacenamiento, cada uno con diferentes características en tecnología y velocidad, sin embargo, todos comparten una característica específica, tienen capacidad finita con respecto a la cantidad de información que pueden guardar.

Según (Ramez & Navathe, 2007), en el libro Fundamentos de Sistemas de Bases de Datos, hay una mejora continua en la capacidad de almacenamiento y las velocidades de transferencia asociadas con los discos, además su precio baja progresivamente. Sin embargo, cabe mencionar que se puede aumentar la duración de video con base a tomar decisiones durante la captura de cada fotograma.

(Deepika & Babu, 2014), indican que la video vigilancia tradicional tiene una enorme cantidad de espacio de almacenamiento. Grabar todo lo que la cámara de video vigilancia captura ocupa gran espacio de almacenamiento y por lo tanto limita la duración de video que se puede almacenar.

Un escenario donde podemos ejemplificar el funcionamiento tradicional de sistemas de video vigilancia, es el circuito cerrado de televisión (CCTV) donde las necesidades de almacenamiento son grandes, debido a que guardan toda la información que capturan y muchas veces esta información no es relevante. Esta información se almacena por periodos de 30 días o más, por lo que la necesidad de contar con dispositivos de gran capacidad de almacenamiento es muy alta. Con base en lo anterior es importante guardar videos que contengan personas, esto con la finalidad de aprovechar al máximo la capacidad de almacenamiento de los dispositivos. La utilidad de poder contar con criterios para descartar imágenes que no incluyan personas reducirá considerablemente el tamaño del video.

Basados en el análisis de tecnologías, métodos y sistemas existentes es viable reducir la cantidad de fotogramas de video que se almacenan, mediante la toma de decisiones basadas en la identificación y el reconocimiento facial en tiempo real.

1.2 Propósito

A través del uso de algoritmos de reconocimiento e identificación facial, la presente propuesta plantea filtrar y procesar fotogramas que se capturan a través de una cámara de video, lo cual permitirá utilizar de forma más eficiente el espacio de almacenamiento de cualquier dispositivo electrónico que tenga las características de hardware y software requeridas, evitando así guardar largas secuencias de imágenes estáticas en video.

1.3 Posibles Aplicaciones

La presente propuesta puede tener diversas aplicaciones, por ejemplo, en sistemas que implementen biométricas en puertas de acceso, el modelo permitiría documentar información, como fecha, hora y persona, que ingresó a un área específica de una empresa sin necesidad de que el sistema esté grabando todo el tiempo, solo almacenaría dicha información, cuando una persona sea detectada a través de la cámara de video.

Otra posible aplicación puede ser en los cajeros automáticos de los bancos, donde se podría almacenar información de la persona que realiza una transacción, al colocar una cámara de video arriba de la pantalla del cajero, lo cual permitiría guardar información de la persona que llevó a cabo una transacción y de esta manera poder identificar qué persona, realizó ciertas transacciones específicas en un momento específico.

En sistemas de video vigilancia orientados a identificar personas, se podría implementar un prototipo de esta naturaleza ya que evitaría tener escenas vacías, ya que solo almacenaría información que incluye personas. También se podría implementar en sistemas de vigilancia caseros, a través de un *Smartphone* ya que estos tipos de dispositivos cuentan con la tecnología y las características (cámara de video, procesador y dispositivo de almacenamiento) para implementarlo, lo cual pondría al alcance de todas las personas que cuenten con un teléfono inteligente, sin necesidad de tener que invertir en un sistema de video vigilancia completo.

1.4 Estado del Problema y Selección de Mejores Prototipos

1.4.1 Detección de Objetos

1.4.1.1 Métodos *Modified RA Tracking Module, GMM Tracking Module, CAMshift Module, ST-MRF Module*

El seguimiento de objetos en video basados en escenas estáticas o dinámicas es uno de los problemas más difíciles con una gran variedad de aplicaciones, actualmente es uno de los temas de investigación más activos en visión por computadora (Divya, 2015).

(Divya, 2015), propone cuatro métodos la para detección de objetos en movimiento (Modified RA Tracking Module, GMM Tracking Module, CAMshift Module, ST-MRF Module), donde concluyen que el rendimiento de los métodos no es único, diferentes condiciones como cambios de iluminación, tipo de objeto a rastrear, detección de objetos no

rígidos, movimiento de fondo, entre otras, dificultan la identificación de objetos en movimientos.

El funcionamiento del sistema planteado por (Divya, 2015) se describe en el Diagrama 1

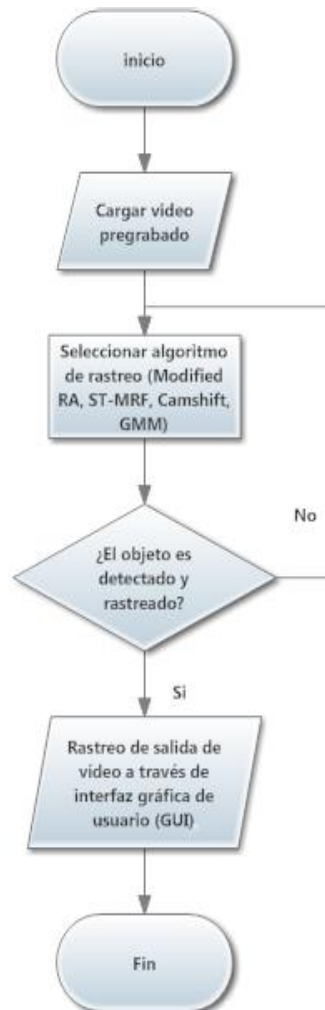


Diagrama 1. Arquitectura del modelo (Modified RA Tracking Module, GMM Tracking Module, CAMshift Module, ST-MRF Module).

La principal ventaja de estos métodos, radica en que pueden detectar objetos en movimiento, lo cual hace flexible el proceso de rastreo pudiéndose adaptar a búsquedas de objetos específicos.

Por otro lado estos cuatro métodos poseen ciertas desventajas, por ejemplo, el tiempo de procesamiento el cual se puede observar en la Tabla 1, esta limitante no permite detectar objetos en tiempo real. Otra desventaja sería que requiere precarga de un video el cual se descompondrá en fotogramas para su posterior procesamiento.

Tabla 1. Tiempos de procesamiento.

Método	Tiempo de procesamiento en segundos
Modified RA	12.291505
MMS tracking	100.420769
GMM tracking	56.652406
ST-MRF tracking	67.567901

1.4.1.2 AdaBoost - Algoritmo *Viola Jones*

El algoritmo *Viola Jones* tiene un gran rendimiento, gracias al concepto de imagen integral, la cual, es una representación intermedia para la imagen de entrada, lo que permite realizar las operaciones sobre la imagen de forma muy rápida, de este manera puede realizar detección de rostros en tiempo real. A continuación se describen características del hardware y arquitectura del modelo utilizado por (Viola & Jones, 2001):

- Imagen de entrada de 384 x 288 pixeles.
- Detección de rostros a 15 fotogramas por segundo.
- Procesador 700 MHz.

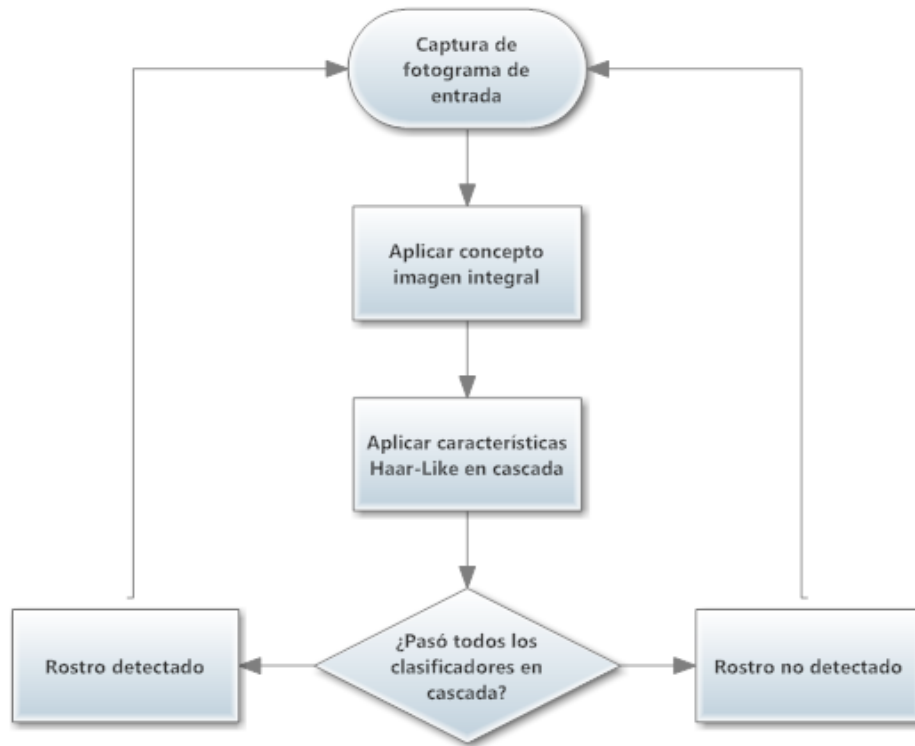


Diagrama 2. Arquitectura AdaBoost.

Los resultados obtenidos por (Viola & Jones, 2001) se muestran en Tabla 2. Como podemos observar la tasa de detección del algoritmo es muy eficiente con respecto a otros detectores.

Tabla 2. Datos comparativos de tasas de detección de falsos positivos en el conjunto de detección MIT+CMU.

Falsas detecciones	10	31	50	65	78	95	167
Detector							
Viola-Jones	76.1%	88.4%	91.4%	92.0%	92.1%	92.9%	93.9%
Viola-Jones (voting)	81.1%	89.7%	92.1%	93.1%	93.1%	93.2%	93.7%
Rowley-Baluja-Kanade	83.2%	86.0%	-	-	-	89.2%	90.1%
Schneiderman-Kanade	-	-	-	94.4%	-	-	-
Roth-Yang-Ahuja	-	-	-	-	(94.80%)	-	-

1.4.2 Sistemas de Video Vigilancia

1.4.2.1 Operador de Sobel

(Suradkar, Kolte, Jamdade, & Gokhale, 2015), mencionan que las actuales técnicas en sistemas de video vigilancia, guardan el video completo incluyendo fotogramas ociosos, generando una gran necesidad de almacenamiento. Proponen una aplicación para ignorar los fotogramas ociosos con un sistema efectivo de detección de objetos en tiempo real, con la ayuda del algoritmo de *Sobel*.

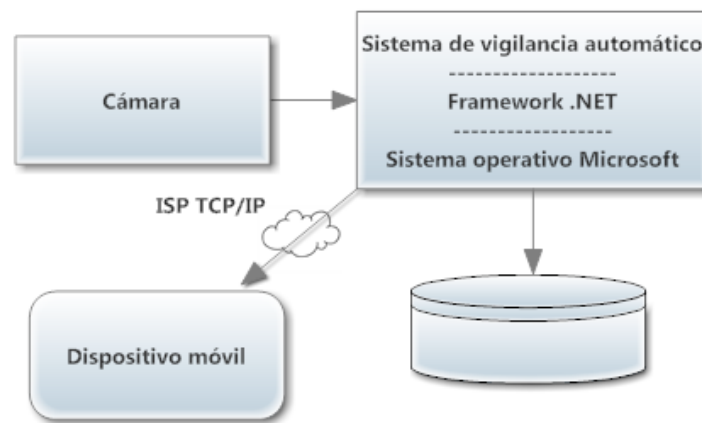


Diagrama 3. Arquitectura del modelo propuesto por (Suradkar et al., 2015).

El operador *Sobel* realiza una medición 2-D del gradiente espacial en una imagen, haciendo énfasis en las regiones de alta frecuencia espacial que corresponde a los bordes. Normalmente se utiliza para encontrar el aproximado absoluto de la magnitud del gradiente en cada punto de una imagen de entrada en escala de grises. Este algoritmo se basa en la diferencia absoluta así como la combinación de regiones. Las regiones activas se obtienen a través de la diferencia de fotogramas y seleccionando un efectivo valor de umbral. La diferencia absoluta se calcula comparando el fotograma actual con el fotograma anterior de la secuencia de video.

Este algoritmo es eficiente para la detección de objetos móviles en aplicaciones de video vigilancia.

El operador de *Sobel* realiza una convolución que suaviza la imagen de entrada y hace que el operador sea menos sensible al ruido.

La principal desventaja de este modelo, es que está orientado a tomar decisiones con base en la detección del movimiento en las secuencias de video y la identificación del contorno de objetos, lo cual es una limitante, ya que se tendría que realizar un desarrollo adicional para la detección de rostros. (Suradkar et al., 2015) mencionan que la principal ventaja del operador de *Sobel* es la relativa facilidad para su implementación.

1.3.2.1 Modelo *Motion Detection In Real-Time*

Para resolver el problema de grabar todo lo que las cámaras capturan (Deepika & Babu, 2014), proponen un robusto algoritmo de detección de movimiento en tiempo real, el cual permite identificar movimiento en la escena de esta forma solo almacenarán los fotogramas que tienen movimiento.

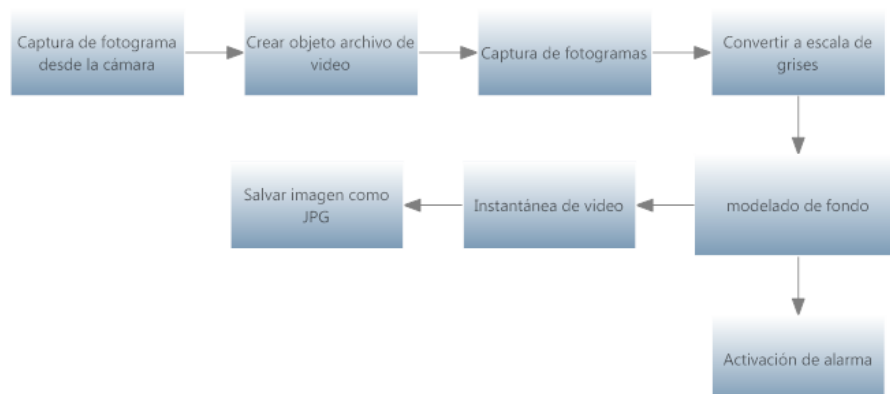


Diagrama 4. Arquitectura del modelo *Motion detection in real time*.

El primer paso del modelo que proponen (Deepika & Babu, 2014), es realizar un pre-procesado de la imagen para remover el ruido y hacer una conversión de color, posteriormente se lleva a cabo el modelado de fondo, para esto se utiliza el algoritmo de detección automática de objetos en movimiento. Las regiones en movimiento se obtienen automáticamente a través de la diferencia de fotogramas, con un algoritmo de selección del umbral. El siguiente modulo, es el núcleo del sistema de video vigilancia que incluye principalmente la identificación de objetos, la cual permite la activación de la alarma. Finalmente se hace la compresión de video en codificación JPEG.

Las desventajas de este modelo, es principalmente la dificultad de la detección de objetos con sombras y la gran cantidad de recursos de almacenamiento, que se requiere para almacenar el video.

Muchos sistemas, son capaces de realizar un seguimiento fiable del movimiento humano, en ambientes controlados interiores y exteriores. Sin embargo, la detección de objetos y seguimiento en la noche, sigue siendo un gran problema para video vigilancia. Los objetos son a menudo pequeños y tienen bajo contraste, con respecto al fondo. Los métodos tradicionales basados en el análisis entre la diferencia de sus fotogramas sucesivos no son efectivos.(Huang, Wang, Tan, & Maybank, 2008).

Existen dos problemas fundamentales en visión artificial: detección de contornos y segmentación de imágenes (Arbelaez, Maire, & Fowlkes, 2011). Proponen un algoritmo para detección de contorno, el cual, combina múltiples señales locales en un framework de globalización basado en la clusterización (agrupación) espectral. El algoritmo de segmentación, consiste en una maquinaria genérica para transformar la salida de cualquier

detector de contorno en un árbol de regiones jerárquico. De esta manera reducen el problema de la segmentación de la imagen a la detección del contorno.

(Levinshtein, Sminchisescu, & Dickinson, 2010) proponen la detección del cierre de contorno, es decir, la búsqueda de fragmentos de ciclos de contorno desconectados que separa un objeto de su fondo, es un problema importante en la percepción de agrupación. Buscar el espacio completo de posibles agrupaciones es muy complejo, enfoques anteriores han adoptado poderosas heurísticas de agrupación de percepción, tales como, la proximidad y la co-curvinalidad para gestionar la búsqueda. Introdicen una nueva formulación del problema transformando el problema de búsqueda de fragmentos de ciclos de contorno, en encontrar subconjuntos de súper píxeles cuyo límite colectivo tiene un fuerte contorno de soporte en la imagen.

Tomando en cuenta los datos obtenidos en la Tabla 3 donde se analizaron y compararon características de diferentes propuestas y modelos, se seleccionaron como base para la investigación los siguientes prototipos:

- AdaBoost
- Modelos *Modified RA Tracking Module*, *GMM Tracking Module*, *CAMshift Module*, *ST-MRF Module*
- Motion Detection In Real-Time

Tabla 3. Análisis comparativo de características de los modelos revisados.

Modelo Características	RA, ST-MRF, Camshift	AdaBoost	Operador de Sobel	PCA, wavelets with SVM	Motion Detection In Real-Time	Face-TLD
Cámara estática	No	X	Si	No	Si	No
Video en tiempo real	No	Si	Si	No	Si	Si
Identificación movimiento	No	No	Si	No	Si	No
Identificación objetos	Si	Si	No	No	No	No
Detección facial	No	Si	No	No	No	Si
Identificación rostros	No	Si	No	Si	No	No
Base de conocimiento (rostros)	No	Si	No	Si	No	Si
Almacenamiento en video	No	No	Si	No	Si	No
Tasa de detección (alta, media, baja)	Baja	Alta	Media	Alta	Media	Alta
Tiempo de proceso (Bueno, regular, malo)	Malo (8s)	Bueno	X	X	X	X
Detección de contornos	X	No	Si	X	No	No
Detección de rasgos	X	Si	No	Si	No	X
Rendimiento(bueno, regular, malo)	Malo	Bueno	Regular	Bueno	Bueno	Regular
Requiere entrenamiento	No	No	No	Si	No	Si
Referencias	(Divya, 2015)	(Viola & Jones, 2001)	(Suradkar et al., 2015)	(Gumus et al., 2010)	(Deepika & Babu, 2014)	(Kalal et al., 2010)

1.5 Aportaciones

La principal aportación de la presente tesis, está orientado a reducir y administrar la cantidad de fotogramas que se guardan en dispositivos de almacenamiento secundario, con la información que se captura a través de cámaras en ambientes estáticos.

Haciendo una estimación, un sistema de video vigilancia pequeño cuenta desde 1 a 30 cámaras, que capturan en video 24 cuadros por segundo, con un tamaño de 640 x 480 pixeles, suponiendo que el sistema funciona las 24 horas del día, considerando un peso de 10 Kb por cada cuadro, al final de un periodo de 60 días, la cantidad de almacenamiento requerido por cada cámara es de aproximadamente 1,182 Gb.

Con estos datos podemos concluir, que un sistema pequeño con 30 cámaras de video de baja resolución requiere al menos 35,460 GB de almacenamiento (34.62 TB). Si consideramos imágenes en alta resolución, las necesidades de hardware para almacenamiento aumentan considerablemente, lo que se traduce en altos costos de operación.

1.6 Objetivos

1.6.1 Objetivo General

En el presente trabajo de investigación, se propone desarrollar un nuevo enfoque para sistemas de video vigilancia, basado en criterios de identificación y reconocimiento facial en tiempo real, que permitan filtrar y reducir la cantidad de información almacenada en video.

1.6.2 Objetivos Específicos

- Estudio y análisis del estado actual del problema.
- Selección de prototipos y análisis de su desempeño.

- Evaluación de métodos y algoritmos de sistemas existentes.
- Diseño de solución.
- Implementar un prototipo para captura de video en el cual se detectarán e identificarán personas.
- Definir una base conocimientos de rostros.
- Utilizar la base de conocimiento para identificar los rostros en el video.
- Almacenar una secuencia de 10 imágenes por minuto para rostros incluidos en la base de conocimiento.
- Almacenar una secuencia de 10 imágenes por segundo para rostros no incluidos en la base de conocimiento.
- Documentación de las pruebas y evaluación del prototipo

1.7 Alcances y Limitaciones

Se espera desarrollar un prototipo que utilice reconocimiento facial, para reducir la cantidad de fotogramas almacenados en video.

Se capturarán imágenes en secuencia a través de una cámara de video, las cuales se procesarán para detectar e identificar un rostro utilizando una librería de INTEL para visión artificial.

Una vez que se detecte un rostro, se comparará con una base de conocimiento, si el rostro es identificado, se almacenará un secuencia de 10 imágenes por minuto, en caso de no ser identificado se almacenará una secuencia de 10 imágenes por segundo.

No se detectarán rostros en la oscuridad. El sistema tendrá en la base de conocimiento un máximo de 5 caras las cuales estarán en escala de grises. La distancia entre la cámara y el objeto va de un rango de 0.1 a 1.5 metros. Se considera una habitación cerrada, con buena iluminación para la detección e identificación facial. El formato de almacenamiento en video es AVI. Se considera un ángulo de detección del rostro de 10 grados. Se capturarán cuadros de 640 x 480, 24 cuadros por segundo

1.8 Hipótesis

El modelo propone, reducir la cantidad de información almacenada en video a través de tecnologías de reconocimiento facial, las cuales permitirán analizar y procesar la información de entrada y poder tomar decisiones, con base en la información contenida en el fotograma.

1.9 Hardware y Software a utilizar

Hardware:

- Equipo de cómputo HP Pavilion dv4, procesador AMD Turion 2.20 GHz, 4 GB en RAM, disco duro 284 GB
- Webcam Low-light, VGA camera, fixed (no tilt) with activity light, 640×480 by 24 frames per second.

Software:

- Windows 7 a 64 bits
- Visual Studio .NET

- OpenCV librería de INTEL para visión artificial.
- Librería AForge.NET para almacenamiento en video.