



INSTITUTO POLITÉCNICO NACIONAL

**ESCUELA SUPERIOR DE INGENIERÍA
MECÁNICA Y ELÉCTRICA
UNIDAD ZACATENCO**

**SECCIÓN DE ESTUDIOS DE POSGRADO
E INVESTIGACIÓN**

**“Detección de movimiento de objetos mediante
secuencias de video”**

TESIS

Que para obtener el grado de

**MAESTRO EN CIENCIAS EN INGENIERÍA
ELÉCTRÓNICA**

Presenta:

Ing. José Portillo Portillo

Directores de Tesis:

Dr. Francisco Javier Gallegos Funes

Dra. Ivonne Bazán Trujillo

México D.F., Mayo de 2012

ÍNDICE GENERAL

ÍNDICE GENERAL.....	i
RESUMEN.....	ii
ABSTRACT.....	iii
ÍNDICE DE CONTENIDO.....	iv
ÍNDICE DE FIGURAS.....	vi
ÍNDICE DE TABLAS.....	vii
ABREVIATURAS Y NOMENCLATURAS.....	viii

RESUMEN

En este trabajo se presenta un nuevo método para la detección de movimiento en secuencias de video, mediante el análisis de las características locales basados en estadística de orden, usando la Diferencia Absoluta Mediana (MAD). Durante el desarrollo e implementación de la propuesta fue necesario crear umbrales de la MAD para discriminar movimiento no significativo. También se desarrollaron técnicas de pre-procesamiento de imágenes que mejoran el desempeño de la propuesta, una vez que se logro la detección de movimiento se realizaron técnicas de Morfología Matemática para identificar si existía mas de un objeto moviéndose en la escena. Los objetos detectados fueron identificados mediante el área de movimiento producida, para esto se desarrollo un umbral que prioritariamente señale al objeto de mayor área de movimiento. El algoritmo desarrollado se probó en secuencias de video con ambientes interiores y exteriores, mostrando el desempeño y robustez del algoritmo al discriminar el movimiento no significativo.

ABSTRACT

This work addresses a new method for motion detection in video sequences, by analyzing local characteristics based on order statistics, using the Median Absolute Deviation (MAD). During the development and implementation of the proposal it was necessary to build up the MAD thresholds so as to eliminate non-significant movement. Furthermore, Image pre-processing techniques were developed so as to enhance the performance of the algorithm, once the motion detection was achieved, Mathematical Morphology techniques were developed to identify whether there was more than one object in motion. The detected objects were identified through the area produced by the motion. For this reason, thresholds were set to trace and prioritize objects that covered greater areas. The developed algorithm was tested on video sequences upon indoor and outdoor environments, and this proved the high performance and robustness of the algorithm.

ÍNDICE DE CONTENIDO

Capítulo 1: Introducción al proyecto	1
1.1 Introducción	1
1.2 Planteamiento del problema.....	1
1.3 Justificación	1
1.4 Objetivos.....	2
1.5 Organización de la Tesis	2
1.6 REFERENCIAS BIBLIOGRÁFICAS.....	2
CAPITULO 2 ESTADO DEL ARTE: DETECCION DE MOVIMIENTO Y SISTEMAS DE VIGILANCIA VISUAL.....	4
2.1 Introducción	4
2.2 El sistema de vigilancia visual.....	4
2.3 Modelado de la escena y extracción de imágenes de fondo.....	5
2.4 Segmentación del Movimiento	6
2.5 Clasificación de objetos	7
2.6 Seguimiento del objeto.	8
2.7 Descripción y comprensión del comportamiento	10
2.8 Identificación de personas por sistemas de vigilancia visual.	11
2.9 Fusión de datos de múltiples cámaras	12
2.10 Secuencias de video en espacio RGB y escala de grises.	12
2.11 Uso de la MAD como estimador de la homogeneidad de las regiones de una trama	14
2.12 Conclusiones.....	16
2.13 REFERENCIAS BIBLIOGRÁFICAS.....	16
Capítulo 3: METODOS y algoritmos PROPUESTOS	21
3.1 INTRODUCCIÓN.....	21
3.2 ALGORITMO A PARA SECUENCIAS DE VIDEO SIMULADAS.....	21
3.3 ALGORITMO B : IMPLEMENTACION DE UN UMBRAL DINAMICO PARA DISCRIMINAR MOVIMIENTO NO SIGNIFICATIVO EN SECUENCIAS DE VIDEO REALES.	23

3.4	IMPLEMENTACION DE TECNICAS DE PREPROCESAMIENTO EN LAS TRAMAS Y EL CAMBIO DEL TAMAÑO EN LAS VENTANAS DE ANALISIS.	29
3.5	ALGORITMO D: IMPLEMENTACION DEL ALGORITMO KNN Y OPERACIONES MORFOLOGICAS	40
3.6	CONCLUSIONES	49
3.7	REFERENCIAS BIBLIOGRÁFICAS	50
CAPÍTULO 4 RESULTADOS OBTENIDOS, CONCLUSIONES Y TRABAJO FUTURO.		52
4.1	INTRODUCCIÓN	52
4.2	CONCLUSIONES	68
4.3	TRABAJO FUTURO.....	68
4.4	REFERENCIAS BIBLIOGRÁFICAS.....	69
CAPÍTULO 5 PARTICIPACIÓN EN CONGRESOS		70
5.1	CONGRESOS NACIONALES	70
5.1	CONGRESOS INTERNACIONALES	70

ÍNDICE DE FIGURAS

FIGURA 2. 1	ESTRUCTURA GENERAL DE UN SISTEMA DE VIGILANCIA VISUAL	5
FIGURA 2. 2	SILUETAS HUMANAS EN MOVIMIENTO USANDO FORMAS GEOMÉTRICAS BÁSICAS.....	7
FIGURA 2. 3	MODELO GEOMÉTRICO DEL HEMISFERIO SUPERIOR DEL CUERPO HUMANO.	8
FIGURA 2. 4	DIAGRAMA DE UNA SECUENCIA DE VIDEO EN ESPACIO RGB	13
FIGURA 2. 5	ESPACIO TRIDIMENSIONAL RGB	13
FIGURA 2. 6	DIAGRAMA DE UNA SECUENCIA DE VIDEO EN ESCALA DE GRISES.	14
FIGURA 3. 1	DIAGRAMA DE FLUJO DEL <i>ALGORITMO A</i>	22
FIGURA 3. 2	SECUENCIA DE VIDEO SIMULADA CON UN OBJETO EN MOVIMIENTO EN DIRECCIÓN HORIZONTAL.	23
FIGURA 3. 3	DIAGRAMA DE FLUJO DEL <i>ALGORITMO B</i>	24
FIGURA 3. 4	A) TRAMA ANTERIOR: TRAMA 10 EN LA SECUENCIA DE VIDEO "CHAIR", B) TRAMA ACTUAL: TRAMA 11 EN LA SECUENCIA DE VIDEO "CHAIR", C) MODIFICACIÓN DEL CONTRASTE EN LAS VENTANAS DE ANÁLISIS BASADOS EN EL ARREGLO RIJ DEL <i>ALGORITMO A</i>	25
FIGURA 3. 5	VENTANAS DE ANÁLISIS (6,2) A) TRAMAS 11 (ACTUAL), B) 10 (ANTERIOR) CON LOS VALORES Y EL NIVEL DE GRIS QUE REPRESENTAN, EL RESULTADO DEL CÁLCULO DE LA <i>MADA(6,2)</i> , LA <i>MADB(6,2)</i> Y EL RESULTADO <i>R(6,2)</i>	26
FIGURA 3. 6	HISTOGRAMA DE LAS RESTAS DE LAS <i>MAD</i> PARA LAS TRAMA 10 Y 11 DEL VIDEO "CHAIR"	27
FIGURA 3. 7	DISTRIBUCIÓN NORMAL DE LAS RESTAS DE LAS <i>MAD</i> Y LOS VALORES OBTENIDOS PARA LOS LIMITES POSITIVO Y NEGATIVO EN LAS TRAMAS 10 Y 11 DEL VIDEO "CHAIR".....	28
FIGURA 3. 8	RESULTADOS OBTENIDOS CON LA APLICACIÓN A) ALGORITMO <i>A</i> Y B) ALGORITMO <i>B</i> A LAS TRAMAS 11 Y 10 DE LA SECUENCIA DE VIDEO "CHAIR".	29
FIGURA 3. 9	PLANOS DE BITS	30

FIGURA 3. 10 PLANOS DE BITS DE LA TRAMA 35 DEL VIDEO "CHAIR", A) PLANO 0, B) PLANO 1, C) PLANO 2, D) PLANO 3, E) PLANO 4, F) PLANO 5, G) PLANO 6, H) PLANO 7, I) IMAGEN ORIGINAL.....	32
FIGURA 3. 11 IMÁGENES GENERADAS POR LA SUMA O RESTA DE LOS PLANOS DE BITS. A) IMAGEN ORIGINAL MENOS EL PLANO 0, B) IMAGEN ORIGINAL MENOS LOS PLANOS 0 Y 1, C) IMAGEN ORIGINAL SIN LOS PLANOS 0,1 Y 2, D) IMAGEN ORIGINAL MENOS LOS PLANOS 0,1,2 Y 3, E) IMAGEN GENERADA POR LA SUMA DEL PLANO 5, 6 Y 7, F) IMAGEN GENERADA POR LA SUMA DE LOS PLANOS 6 Y 7.....	33
FIGURA 3. 12 A) DESPLAZAMIENTO DE UNA BARRA EN EL TIEMPO T, B) DESPLAZAMIENTO DE UNA BARRA EN EL TIEMPO T + ΔT QUE MUESTRA EL PRINCIPIO BÁSICO DEL "PROBLEMA DE APERTURA"	33
FIGURA 3. 13 PROBLEMA DE APERTURA EN UNA LA TRAMA 25 DEL VIDEO "CHAIR"	34
FIGURA 3. 14 MECANISMO DE FILTRADO ESPACIAL.	35
FIGURA 3. 15 MOVIMIENTO DE LOS COEFICIENTES DE LA VENTANA Y LOS COEFICIENTES DE LA VECINDAD CON RESPECTO AL PIXEL CENTRAL F, C DE LA IMAGEN ORIGINAL.....	36
FIGURA 3. 16 RESULTADOS INDIVIDUALES Y COMBINADOS DE LOS PRE-PROCESAMIENTOS PROPUESTOS. A) IMAGEN ORIGINAL, B) IMAGEN ORIGINAL MENOS LOS PLANOS 0, 1, 2 Y 3, c) IMAGEN EMBORRONADA, D) ENMASCARAMIENTO DE LOS 4 LSB + EMBORRONAMIENTO, E) IMAGEN EMBORRONADA + ENMASCARAMIENTO DE LOS 4 LSB.	37
FIGURA 3. 17 DIAGRAMA DE FLUJO DEL ALGORITMO C.....	38
FIGURA 3. 18 A) IMAGEN ORIGINAL, B) RESULTADO DEL ALGORITMO A, c) RESULTADO DEL ALGORITMO C, D) RESULTADO DEL ALGORITMO D. LOS ALGORITMOS FUERON APLICADOS A LAS TRAMAS 11 Y 10 DE LA SECUENCIA DE VIDEO "CHAIR"	39
FIGURA 3. 19 A) TRAMA 145 EN ESPACIO RGB. RESULTADOS DEL ALGORITMO C APLICADO A LAS TRAMAS 144 Y 145 DE LA SECUENCIA DE VIDEO "LAURA_EXTERIOR" UTILIZANDO: B) VENTANAS DE ANÁLISIS DE 8x8 PÍXELES, C) VENTANAS DE ANÁLISIS DE 8x4 PÍXELES, D) VENTANAS DE ANÁLISIS DE 4x4 PÍXELES, E) VENTANAS DE ANÁLISIS DE 2x4 PÍXELES.	41
FIGURA 3. 20 DESPLAZAMIENTO DE LAS VENTANAS PARA REALIZAR LA IMPLEMENTACIÓN DEL ALGORITMO KNN PROPUESTO.....	42
FIGURA 3. 21 DIAGRAMA DE FLUJO DEL ALGORITMO D, IMPLEMENTACIÓN DE UN ALGORITMO KNN	43
FIGURA 3. 22 A) TRAMA 145 DE LA SECUENCIA DE VIDEO "LAURA_EXTERIOR". APLICACIÓN DEL ALGORITMO KNN CON: B) UMBRAL_KNN = 3, c) UMBRAL_KNN = 20, D) UMBRAL_KNN = 7.	44
FIGURA 3. 23 ELEMENTO ESTRUCTURAL USADO PARA LAS OPERACIONES MORFOLÓGÍAS, UNA LÍNEA DE 15 PÍXELES DE LONGITUD, 1 PÍXEL DE ANCHO Y 45 GRADOS DE INCLINACIÓN.	46
FIGURA 3. 24 A) TRAMA 145 EN EL ESPACIO RGB, B) MODIFICACIÓN DEL CONTRASTE PARA SEÑALAR EL MOVIMIENTO DETECTADO, C) IMAGEN BINARIA DEL MOVIMIENTO, D) IMAGEN BINARIA DEL MOVIMIENTO DESPUÉS DE APLICAR LA OPERACIÓN MORFOLÓGICA DE CERRADURA.....	46
FIGURA 3. 25 A) TRAMA 145 EN ESPACIO RGB, B) DIVERSOS OBJETOS EN MOVIMIENTO, C) APLICACIÓN UMBRALÁREA.	48
FIGURA 3. 26 ENMARCADO Y CENTROIDE DEL OBJETO EN MOVIMIENTO, A) EN LA IMAGEN CON MODIFICACIÓN DE CONTRASTE, B) EN LA IMAGEN BINARIA DESPUÉS DE LA OPERACIÓN DE CERRADURA, C) EN LA IMAGEN EN ESPACIO RGB.....	49
FIGURA 4. 1 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO "LAURA_PASILLO", DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB. A) TRAMA 180, B) TRAMA 205, C) TRAMA 230, D) TRAMA 250.	58
FIGURA 4. 2 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO "MARIANA_PASILLO", DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA,	

<p>APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB.</p>	
<p>A) TRAMA 115,</p>	
<p>B) TRAMA 135,</p>	
<p>C) TRAMA 157,</p>	
<p>D) TRAMA 191.</p>	59
<p>FIGURA 4. 3 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “LAURA_EXTERIOR”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB.</p>	
<p>A) TRAMA 80,</p>	
<p>B) TRAMA 133,</p>	
<p>C) TRAMA 160,</p>	
<p>D) TRAMA 225.</p>	60
<p>FIGURA 4. 4 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “PROFESOR_EXTERIOR”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB.</p>	
<p>A) TRAMA 67,</p>	
<p>B) TRAMA 98,</p>	
<p>C) TRAMA 121,</p>	
<p>D) TRAMA 151.</p>	61
<p>FIGURA 4. 5 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “ALUMNAS_2”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB.</p>	
<p>A) TRAMA 113,</p>	
<p>B) TRAMA 139,</p>	
<p>C) TRAMA 176,</p>	
<p>D) TRAMA 193.</p>	62
<p>FIGURA 4. 6 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “ALUMNOS_3”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB.</p>	
<p>A) TRAMA 285,</p>	
<p>B) TRAMA 310,</p>	
<p>C) TRAMA 330,</p>	
<p>D) TRAMA 510.</p>	63
<p>FIGURA 4. 7 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “CICLISTA”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB.</p>	
<p>A) TRAMA 60,</p>	
<p>B) TRAMA 72,</p>	
<p>C) TRAMA 92,</p>	
<p>D) TRAMA 113.</p>	64

FIGURA 4. 8 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “MOTONETA”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB. A) TRAMA 380, B) TRAMA 395, C) TRAMA 460, D) TRAMA 501.	65
FIGURA 4. 9 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “VEHÍCULO”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB. A) TRAMA 340, B) TRAMA 365, C) TRAMA 390, D) TRAMA 446.	66
FIGURA 4. 10 RESULTADOS DEL ALGORITMO D APLICADO A LA SECUENCIA DE VIDEO “PASILLO_PRINCIPAL”, DE ARRIBA ABAJO, TRAMA ACTUAL EN ESPACIO RGB, MODIFICACIÓN DEL CONTRASTE EN VENTANAS CONSIDERADAS COMO MOVIMIENTO, IMAGEN BINARIA DE VENTANAS CONSIDERADAS MOVIMIENTO, APLICACIÓN DE LA OPERACIÓN MORFOLÓGICA DE CERRADURA, APLICACIÓN DEL UMBRAL DE ÁREA MÁXIMA, ENMARCADO Y CENTROIDE DEL OBJETO EN LA TRAMA ACTUAL EN ESPACIO RGB. A) TRAMA 140, B) TRAMA 290, C) TRAMA 375, D) TRAMA 438.	67

ÍNDICE DE TABLAS

TABLA 2. 1 OPERACIONES EFECTUADAS PARA EL CÁLCULO DE LA MAD	15
TABLA 2. 2 OPERACIONES EFECTUADAS PARA EL CÁLCULO DE LA MAD, MOSTRANDO LA INSERCIÓN DE UN DATO ATÍPICO	16
TABLA 3. 1 VALORES BINARIOS Y DECIMALES PARA REALIZAR LA OPERACIÓN LÓGICA <i>AND</i> DE LOS PLANOS DESEADOS	31

ABREVIATURAS Y NOMENCLATURAS

(f,c,t)	Posición de la Fila, Columna y espacio Temporal de un pixel en una secuencia de video.
$\alpha(f,c)$	Posición de la Fila y Columna de un pixel en escala de grises de la imagen original.
$\beta(f,c)$	Posición de la Fila y Columna de un pixel en escala de grises de la imagen modificada.
Γ	Operación de transformación de intensidad.
η	Índice máximo de filas para las ventanas de las operaciones espaciales.
φ	Índice máximo de columnas para las ventanas de las operaciones espaciales.
$\omega(o,p)$	Índices de filas y columnas de las ventanas de operación.

A_{ij}	Elementos de la $i - \text{ésima}, j - \text{ésima}$ ventana de análisis de la trama actual
B_{ij}	Elementos de la $i - \text{ésima}, j - \text{ésima}$ ventana de análisis de la trama anterior
$\lim_{(-)}$	Límite negativo del umbral de la MAD
$\lim_{(+)}$	Límite positivo del umbral de la MAD
MAD_{Aij}	Valor de la MAD de los elementos de la $i - \text{ésima}, j - \text{ésima}$ ventana de análisis de la trama actual.
MAD_{Bij}	Valor de la MAD de los elementos de la $i - \text{ésima}, j - \text{ésima}$ ventana de análisis de la trama anterior.
MAD	Diferencia Absoluta de la Mediana (Median Absolute Deviation)
MED	Mediana
$m \times n$	m-filas de la ventana de análisis, n-columnas de las ventanas de análisis
\bar{r}	Valor promedio del resultado de las operaciones de la MAD
r_{σ}	Desviación estándar del resultado de las operaciones de la MAD
r_{ij}	Resultados de la MAD de la $i - \text{ésima}, j - \text{ésima}$ posición de la ventana de análisis.
u	Número de filas de las ventanas para las operaciones espaciales
v	Número de columnas de las ventanas para las operaciones espaciales

CAPITULO 1: INTRODUCCIÓN AL PROYECTO

1.1 INTRODUCCIÓN

El constante crecimiento de la sociedad, la urbanización y concentración de la población en grandes ciudades implica importantes desafíos para garantizar la seguridad u optimizar los sistemas de transporte en base al comportamiento de los usuarios, una de las tecnologías usadas para afrontar estos desafíos son los sistemas de visión por computadora o vigilancia visual, donde los propósitos fundamentales son: Disuadir, prevenir y detectar actos delictivos o supervisar diversas áreas de instalaciones gubernamentales, industriales y militares para detectar de forma oportuna el acceso de personal no autorizado, detectar incidentes o conductas anormales de las persona o vehículos [4]. También es necesario contar con medios digitales de almacenamiento para deslindar responsabilidades o como evidencia de los hechos.

Para satisfacer los desafíos expuestos, los sistemas de vigilancia visual son un tópico activo en el campo de la investigación, desarrollo y consolidación en la vida cotidiana [1], y dado que los sistemas de vigilancia se basan en los dispositivos de adquisición de imágenes y dispositivos de procesamiento de imágenes, el desarrollo de los sistemas de vigilancia esta ligado al desarrollo de estos dispositivos [3], sin embargo la creación y mejora de los dispositivos, a dado lugar a un crecimiento de las teorías y métodos necesarios para implementar mejores sistemas de vigilancia visual.

1.2 PLANTEAMIENTO DEL PROBLEMA.

Cuando se colocan video cámaras con fines de vigilancia y seguridad se emplean diversas soluciones, una de ellas consiste en desplegar sobre un conjunto de pantallas los escenarios de las áreas que se están vigilando, con las imágenes mostradas el personal de seguridad determina si existe algún incidente, comportamiento anormal o el desarrollo de un delito, si el personal de seguridad lo considera necesario, puede mover la cámara, ampliar la imagen o efectuar alguna operación que le permita corroborar si el evento al que se enfrenta requiere de alguna acción directa [2].

Sin embargo, existe la posibilidad de que el evento no sea detectado por el personal de seguridad, pues observar las imágenes de las áreas vigiladas durante periodos de tiempo prolongados, resulta monótono y por tanto algunos eventos pueden pasar desapercibidos para el personal de seguridad, además la gran cantidad de pantallas que deben ser vigiladas en busca de comportamientos anómalos, disminuye la confiabilidad del sistema de vigilancia [5].

La implementación de sistemas de vigilancia que alerten al personal de seguridad sobre alguna anomalía en la escena, refuerza la efectividad del sistema, para ello se requiere de una serie de etapas que fundamentalmente consisten en determinar si algún objeto se ha desplazado dentro de una escena, etapas posteriores se encargaran de determinar si el objeto detectado es una persona, vehículo o mascota, además establecer si el comportamiento del objeto es anormal y por tanto emitir una alarma [1], [5].

De manera conjunta el sistema puede realizar un seguimiento del objeto resaltando su ubicación en la imagen y de este modo el personal de seguridad tendrá información suficiente para realizar las acciones necesarias

1.3 JUSTIFICACIÓN

Una de las etapas más importantes de los sistemas de vigilancia visual es la detección de movimiento en secuencias de video, existen diversas técnicas de detección que consiste en analizar los cambios entre una escena y otra de pixel a pixel, o la detección de bordes y su comparación entre las escenas, una de las posibles soluciones es la detección de movimiento basado en las características de las regiones de la imagen [1].

La descripción de las características de las regiones en una imagen puede ser efectuada mediante estimadores estadísticos de la intensidad de los píxeles que componen esa región, sin embargo los métodos estadísticos clásicos presentan un bajo desempeño en presencia de valores de intensidad atípicos. Por tanto en este trabajo se propone el uso de un estimador estadístico robusto para caracterizar las regiones de la imagen y de esta manera realizar la detección de movimiento.

1.4 OBJETIVOS

- General

Detectar movimiento de personas en secuencias de video, comparando las características estadísticas locales entre tramas consecutivas.

- Particulares

Proponer e implementar un método basado en el análisis de la estimación de las características estadísticas locales usando la Diferencia Absoluta de la Mediana.

Realizar el seguimiento de la persona en movimiento en la secuencia de video.

1.5 ORGANIZACIÓN DE LA TESIS

La tesis esta dividida en cuatro capítulos, que muestran el desarrollo del método de detección de movimiento basado en el análisis de las características locales, desde la propuesta hasta el desarrollo de nuevas etapas para mejorar el algoritmo hasta mostrar los resultados en diversos ambientes y objetos en las escenas.

- Capítulo 1 En este capítulo realizamos una introducción a las necesidades de implementar un sistema de vigilancia visual y los objetivos de esta tesis
- Capítulo 2 En este capítulo se revisa el estado del arte de las etapas que componen un sistema de vigilancia visual, el concepto de video y tramas, para finalmente enfocarnos en la detección de movimiento basado en el análisis de las características de las regiones de una imagen para proponer el uso de un estimador estadístico robusto denominado Diferencia Absoluta de la Mediana (MAD).
- Capítulo 3 Este capítulo muestra el desarrollo y mejora del algoritmo propuesto, detallando las necesidades que surgen para implementar umbrales de discriminación de movimiento significativo, el pre-procesamiento de imágenes, operaciones morfológicas, umbrales para la detección del objeto de mayor área y finalmente el seguimiento del objeto en la secuencia de video.
- Capítulo 4 En este capítulo se presentan los imágenes de los procesos y resultados finales de aplicar el **Algoritmo D**, en secuencias de video en escenarios interiores, exteriores, con uno o mas objetos en la escena y objetos que se desplazan a velocidades mayores en las que regularmente se desplaza una persona.

1.6 REFERENCIAS BIBLIOGRÁFICAS

[1] H. Weiming, T. Tieniu, W. Liang and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 34(3), 334-352 (2004).

[2] I. Ahmad, Z. He, M. Liao, F. Pereira and M. T. Sun, "Special Issue on Video Surveillance," *Circuits and Systems for Video Technology, IEEE Transactions on* 18(8), 1001-1005 (2008).

[3] X. Li-Qun, "Issues in video analytics and surveillance systems: Research / prototyping vs. applications / user requirements," in *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pp. 10-14 (2007).

[4] R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, and L. Wixson, "A system for video surveillance and monitoring," Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep., CMU-RI-TR-00-12, 2000.

[5] I. Haritaoglu, D. Harwood, and L. S. Davis, "W : Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 809–830, Aug. 2000.

CAPITULO 2 ESTADO DEL ARTE: DETECCION DE MOVIMIENTO Y SISTEMAS DE VIGILANCIA VISUAL.

2.1 INTRODUCCIÓN

En este capítulo tratamos el estado del arte de los sistemas de vigilancia visual, las etapas que comúnmente componen un sistema de vigilancia visual y los métodos desarrollados por los diversos autores para cada etapa, y una vez analizados nos enfocamos en las etapas de Modelado de la escena y Segmentación del movimiento para realizar la propuesta de un método basado en estadística de orden para la detección de movimiento en secuencias de video de cámaras fijas.

2.2 EL SISTEMA DE VIGILANCIA VISUAL

En los últimos años existe notable el incremento en el número de video cámaras para la realización de tareas de vigilancia y monitoreo en entornos domésticos, gubernamentales, industriales y militares. Las tecnologías emergentes y en constante desarrollo permiten realizar tareas de vigilancia visual en escenas dinámicas, que es uno de los temas de investigación más activos en los sistemas de visión por computadora [1], cuenta con prometedoras aplicaciones como el control de acceso en zonas especiales, identificación a distancia de personas, análisis estadístico del flujo de personas y/o vehículos bajo diversos entornos, detección de comportamientos anormales y vigilancia interactiva usando múltiples cámaras [2].

De manera general, los sistemas de video vigilancia en las escenas dinámicas tienen las siguientes etapas: modelado del entorno, detección de movimiento, clasificación de los objetos en movimiento, seguimiento, descripción y comprensión del comportamiento, identificación de personas, y fusión de los datos de múltiples cámaras [1]. La Figura 2.1 muestra un esquema de la estructura general de los sistemas de vigilancia visual.

Casi todos los sistemas de vigilancia visual inician con la detección de movimiento, esta tarea consiste en segmentar regiones que corresponden a los objetos en movimiento del resto de una imagen. Las siguientes tareas como el seguimiento de objetos, la descripción y clasificación de los objetos, dependen en gran medida de la detección de movimiento [3]. Usualmente la detección de movimiento implica el modelado del entorno o escena. La construcción activa y la actualización de los modelos del entorno o escena son indispensables para la vigilancia visual, pues nos permite discriminar entre movimientos significativos y no significativos.

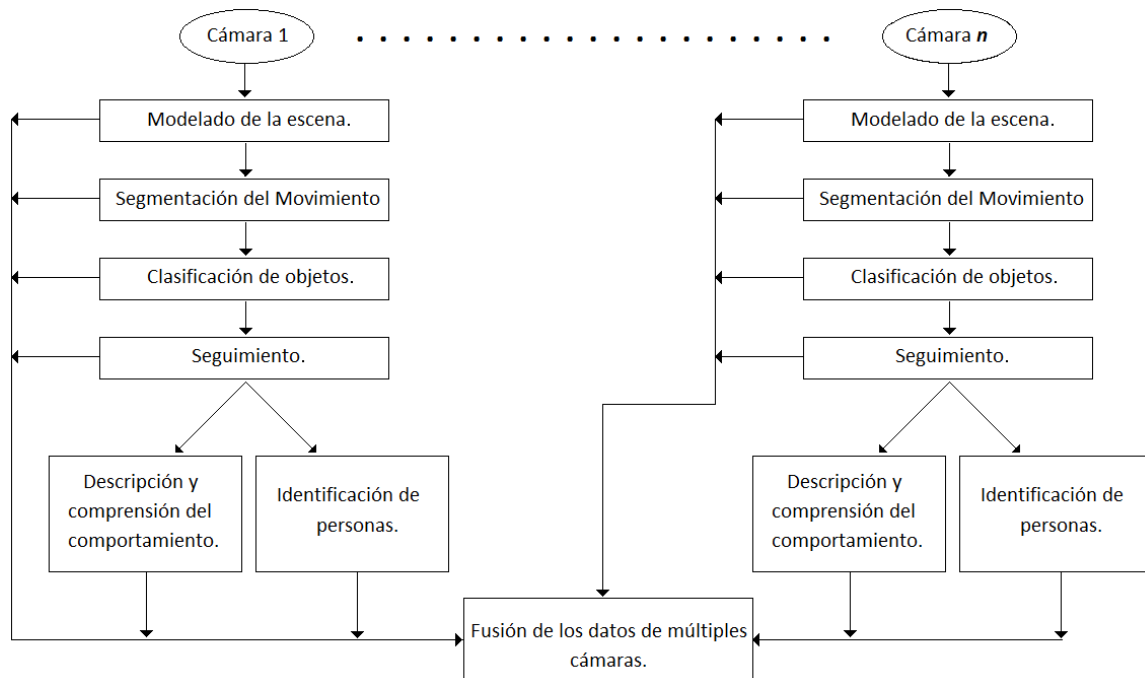


Figura 2. 1 Estructura general de un sistema de vigilancia visual

2.3 MODELADO DE LA ESCENA Y EXTRACCIÓN DE IMÁGENES DE FONDO.

Para las cámaras fijas, el problema clave consiste en obtener y actualizar automáticamente las imágenes consideradas fondo o sin movimiento, a partir de una secuencia de imágenes o tramas.

Factores desfavorables, como la variación de la iluminación, las sombras y movimientos de la naturaleza como pequeños movimientos en hojas y ramas, traen muchas dificultades a la adquisición y actualización de las imágenes de fondo.

Existen varios algoritmos para la solución de estos problemas que incluyen el promedio temporal de una secuencia de imágenes [13, 38], estimación Gaussiana Adaptiva [32], estimación de parámetros basados en procesos de pixeles [36, 37], etc. Por ejemplo, Ridder et al., modelan el valor de cada píxel usando un filtro Kalman para compensar las variaciones de iluminación [38].

Algunos algoritmos proponen una estructura de trabajo para obtener y actualizar las imágenes de fondo basados en un la mezcla de distribuciones Gaussianas (GMM) usadas para cada valor del píxel con el fin de adaptarse a la variación de la iluminación y alteración de los fondos [10, 37]. En el Algoritmo “WallFlower” [40], la obtención y actualización del fondo se realiza en tres niveles: nivel de píxel, nivel región, y el nivel trama. En otros casos, se propone la construcción de un modelo estadístico para presentar cada píxel con tres valores: los valores de intensidad máxima y mínima, y la diferencia de intensidad máxima entre tramas consecutivas observadas durante el período de prueba y actualizando periódicamente estos valores [5].

2.4 SEGMENTACIÓN DEL MOVIMIENTO

La segmentación del movimiento en secuencias de imágenes tiene como objetivo la detección de regiones que corresponden a objetos móviles por ejemplo vehículos y seres humanos. Detectar las regiones de movimiento proporciona las bases para futuras etapas en sistemas de vigilancia visual, por ejemplo, para seguimiento del objeto, análisis del comportamiento, etc. [1, 3]

Algunos métodos generalmente utilizados para la segmentación del movimiento son:

- Substracción del fondo: Es un método popular para la segmentación de movimiento, especialmente en aquellas tramas con un fondo relativamente estático. Se detecta movimiento en regiones de una imagen tomando la diferencia pixel a pixel entre la trama actual y la trama considerada referencia del fondo. Este método es simple pero extremadamente sensible a los cambios en las escenas dinámicas derivadas de la iluminación y pequeños cambios en las escenas. Por lo que, este método requiere un buen modelado de la escena y extracción de imágenes de fondo para reducir la detección de movimiento no significativo [5, 9, 10].

- Diferenciación Temporal: Hace uso de las diferencias de los niveles de intensidad de los píxeles entre dos o tres tramas consecutivas en una secuencia de imágenes para extraer las regiones en movimiento. Este método es muy adaptable a escenas dinámicas, pero en general se obtienen deficientes resultados en la extracción de todos los píxeles correspondientes de movimiento, por ejemplo, es posible falten píxeles en el interior del objeto en movimiento. Un algoritmo de este tipo detecta objetos en movimiento en secuencias de video reales mediante la diferenciación temporal. Después de obtener la diferencia absoluta entre la trama actual y la trama anterior, se utiliza una función de umbral para determinar los cambios. Además, haciendo uso de un análisis de componentes conectados, las secciones de movimiento extraídas, se agrupan en las regiones de movimiento [10]. Una versión mejorada de este método utiliza tres tramas para realizar la diferenciación en lugar de las dos tramas utilizadas originalmente.

- Flujo Óptico: La segmentación de movimiento basada en el flujo óptico utiliza las características de los vectores de flujo de los objetos en movimiento en el tiempo para detectar las regiones en movimiento en una secuencia de imágenes. Por ejemplo, Meyer et al., calculan el desplazamiento del campo vectorial para iniciar un algoritmo de seguimiento basado en el contorno para la extracción de objetos articulados [11, 18]. Los resultados se utilizan para análisis de la forma de caminar. Los métodos basados flujo óptico se pueden utilizar para detectar objetos en movimiento independiente, incluso en presencia de movimiento de la cámara, sin embargo, la mayoría de los métodos de cálculo del flujo óptico son computacionalmente complejos y muy sensibles al ruido, por esta razón, es difícil aplicarlos a secuencias de video en tiempo real sin hardware especializado [12].

Además de los métodos básicos descritos anteriormente, existen algunos otros enfoques para la segmentación de movimiento, Friedman et al. aplica un modelo de distribuciones Gaussianas para clasificar los valores de cada pixel dentro de tres distribuciones predeterminadas y separadas que corresponden a la imagen de fondo, la imagen de movimiento, y la imagen de sombras, también actualiza el componente mixto de forma automática para cada distribución de acuerdo a la probabilidad de membresía [13]. Por lo tanto, los objetos que se mueven lentamente se procesan a la perfección, mientras que las sombras se eliminan de manera más eficaz. El proyecto VSAM (Visual Surveillance and Monitoring) desarrollo con éxito un algoritmo híbrido para la segmentación de movimiento mediante un algoritmo adaptivo de la sustracción de fondo y una técnica de diferenciación usando tres tramas [4]. Stringa propone un nuevo algoritmo

morfológico para detectar el movimiento en las escenas el cual obtiene resultados de segmentación estables, incluso bajo condiciones ambientales variables [14].

2.5 CLASIFICACIÓN DE OBJETOS

Diferentes regiones de movimiento pueden corresponder a diferentes objetos que se mueven en escenarios naturales, por ejemplo, las secuencias de imágenes capturadas por cámaras de vigilancia instaladas en las escenas de tráfico, probablemente incluyen movimiento de seres humanos, vehículos y otros objetos como pájaros que vuelan, las nubes, etc. [1-3]. Para realizar un seguimiento de los objetos de interés y analizar su comportamiento, es fundamental una correcta clasificación de dichos objetos en movimiento. La clasificación de objetos se puede considerar como un tema estándar del área de reconocimiento de patrones. En la actualidad, hay dos categorías principales para la clasificación de objetos en movimiento.

- Clasificación basada en la forma: Las diversas descripciones de la información de la forma de los objetos en movimiento tales como puntos, cajas, siluetas entre otras formas básicas de las regiones de movimiento son usadas para clasificar dichos objetos. Por ejemplo, VSAM considera como características claves: la dispersión, el área, la relación aparente del aspecto, o el cuadro delimitador de las formas geométricas simples [4]. Con estas características, se agrupan las formas geométricas simples en cuatro clases: Una sola persona, vehículos, grupos de personas y aglomeraciones. Kuno et al. hacen uso de los parámetros de las formas geométricas simples correspondientes a los patrones de silueta humana para separar a las personas de otros objetos en movimiento en las escenas procesadas [15]. La Figura 2.2 presenta algunas formas básicas de la figura humana usando líneas y articulaciones, o usando formas geométricas simples.

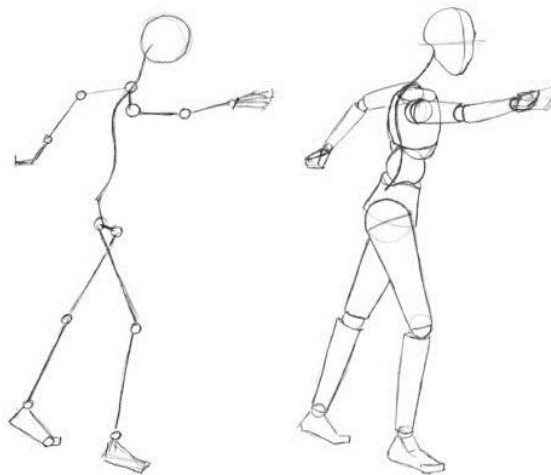


Figura 2. 2 Siluetas humanas en movimiento usando formas geométricas básicas

- Clasificación basada en el movimiento. En general el movimiento humano puede ser caracterizado como un conjunto de barras y articulaciones que muestran características periódicas durante el movimiento, es decir, la forma de caminar de los seres humanos tiende a repetir los mismos movimientos, estas características permiten clasificar si los objetos en movimiento son personas [16], por tanto, siguiendo el objeto de interés y calculando la similitud de los movimientos conforme evolucionan en el tiempo, se puede distinguir el movimiento humano de otros objetos, por ejemplo, del movimiento de vehículos [18]. La Figura 2.3 muestra la caracterización del movimiento humano como un conjunto de barras y articulaciones para el hemisferio superior del cuerpo humano.

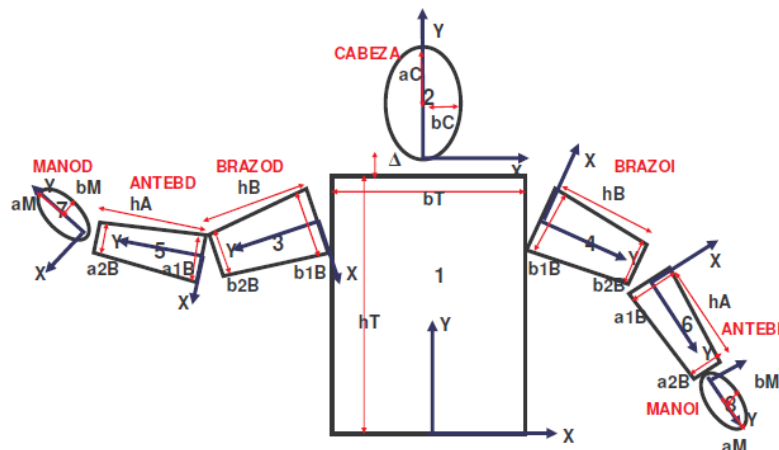


Figura 2. 3 Modelo Geométrico del hemisferio superior del cuerpo humano.

2.6 SEGUIMIENTO DEL OBJETO.

Después de la detección de movimiento, comúnmente los sistemas de vigilancia realizan un seguimiento de dichos objetos de una trama a otra en la secuencia de imágenes. Por lo general, los algoritmos de seguimiento tienen una considerable interacción con los algoritmos de detección de movimiento. Se han desarrollado útiles herramientas matemáticas para los algoritmos de seguimiento, por ejemplo, el filtro de Kalman, el algoritmo denominado Condensación, la red bayesiana dinámica, el método geodésico, etc. Los métodos de seguimiento se dividen en cuatro categorías principales: seguimiento basado en la región, seguimiento basado en el contorno activo, seguimiento basado en las características y seguimiento basado en el modelo [58]. Cabe señalar que esta clasificación no es absoluta ya que los algoritmos de las diferentes categorías se pueden combinar entre si.

- **SEGUIMIENTO BASADO EN LA REGIÓN.** Estos algoritmos siguen los objetos según las variaciones de las regiones de la imagen correspondiente a los objetos en movimiento. La imagen de fondo se actualiza de forma dinámica y las regiones de movimiento se detectan generalmente al restar el fondo de la trama actual [41, 42]. Algunos algoritmos hacen uso de las características de las pequeñas regiones geométricas de una sola persona para realizar el seguimiento en un ambiente interior [8]. El cuerpo humano es considerado como una combinación de algunas figuras geométricas simples que representan varias partes del cuerpo, como la cabeza, el torso y las cuatro extremidades. Mientras tanto, el cuerpo humano y la imagen de fondo se modelan con distribuciones gaussianas obtenidas de los valores de los píxeles. Finalmente, los píxeles que pertenecen al cuerpo humano se asignan a diferentes a las diferentes formas geométricas que representan las partes del cuerpo, utilizando la medida de probabilidad logarítmica, por lo tanto, al realizar el seguimiento de cada forma geométrica simple, el movimiento de las personas es seguido con éxito. Algunas investigaciones proponen métodos adaptivos de substracción de la imagen de fondo en los cuales se combinan el color y el gradiente de la información para hacer frente al problema de las sombras y las señales de color inestables en la segmentación de movimiento [9]. Entonces, el seguimiento se realiza en tres niveles de abstracción: las regiones, las personas y grupos, cada región tiene un cuadro de límite y las regiones se pueden fusionar y/o dividir. Una persona está compuesta de una o más regiones agrupadas bajo la condición de las limitaciones de estructura geométrica en el cuerpo humano, y un grupo de personas se compone de una o más personas agrupadas bajo esta consideración. Por lo tanto,

utilizando el seguimiento de la región y el modelo de apariencia de color individual, se logra un seguimiento perfecto de varias personas, incluso durante la oclusión.

- **SEGUIMIENTO BASADO EN EL CONTORNO ACTIVO.** Los algoritmos de seguimiento basados en el contorno activo siguen los objetos representados mediante la delimitación de sus contornos, además la actualización de dichos contornos se realiza de forma automática entre las tramas sucesivas [6, 33-35]. Estos algoritmos tienen por objeto la extracción directa de las formas de los objetos y proporcionan descripciones de los objetos más eficaces que los algoritmos basados en las regiones. En contraste con los algoritmos de seguimiento basados en la región, los algoritmos basados en contornos activos, describen los objetos de forma más simple y eficaz lo que reduce el costo computacional. Aun cuando las escenas presente alguna perturbación, o los objetos en movimiento presenten oclusión parcial, estos algoritmos, pueden realizar un seguimiento de los objetos de forma continua. Por ejemplo, Paragios et al. detectan y siguen varios objetos en movimiento en secuencias de imágenes mediante una función geodésica objetivo de contorno activo y un esquema de formulación a nivel de conjuntos [22].

- **SEGUIMIENTO BASADO EN LAS CARACTERÍSTICAS:** Estos algoritmos realizan el reconocimiento y seguimiento de objetos mediante la extracción de elementos, agrupados en características de alto nivel y comparando las características entre las tramas, los algoritmos de seguimiento basados en características pueden ser clasificados en tres subcategorías de acuerdo a la naturaleza de las funciones seleccionadas: algoritmos basados en las características globales, algoritmos basados en las características locales y algoritmos basados en la dependencia gráfica.

a) Los algoritmos basados en las características globales incluyen, perímetros, áreas, algunos órdenes de cuadraturas y colores [46, 47], etc. Una persona está delimitada con una caja rectangular cuyo centroide se selecciona como la característica global para el seguimiento [23]. Incluso cuando existe oclusión entre dos personas durante el seguimiento, el movimiento puede ser seguido de manera satisfactoria siempre y cuando la velocidad de los centroides se pueda distinguir con eficacia.

b) Los algoritmos basados en las características locales incluyen segmentos de líneas, segmentos de curvas, vértices [44, 45], etc.

c) Las características usadas en los algoritmos basados en la dependencia gráfica incluyen una variedad de distancias y relaciones geométricas entre las características [43].

Los tres métodos anteriores se pueden combinar entre si, por ejemplo, desarrollando una plantilla activa que determina las características de las regiones y estructuras de un objeto que se construye de forma dinámica basándose en la información de la forma, textura, color y características de los bordes de la región [24].

Los algoritmos de seguimiento basados en las características pueden manejar la oclusión parcial mediante el uso de la información sobre el movimiento del objeto, las características locales y dependencia gráfica. Sin embargo, hay varias deficiencias graves en este tipo de algoritmos, por ejemplo: el nivel de reconocimiento de objetos en 2-D basados en las características de la imagen es baja debido a la distorsión no lineal durante perspectiva de la proyección y las variaciones de la imagen cuando cambia el punto de visión se encuentra en movimiento. Estos algoritmos son generalmente incapaces de recuperar la pose 3-D de los objetos. La estabilidad al tratar con eficacia la oclusión, la superposición e interferencia de las estructuras no relacionadas, es generalmente pobre [1].

- **SEGUIMIENTO BASADO EN EL MODELO:** Los algoritmos de seguimiento basados en modelos siguen los objetos mediante la comparación de modelos de los objetos de interés proyectados y producidos con el conocimiento previo de los datos de la imagen. Estos modelos se construyen generalmente fuera de línea, con mediciones manuales, con herramientas de diseño asistido por computadora o mediante otras técnicas de visión por computadora [1].

El seguimiento de objetos basados en el modelo tienen dos ramas principales que son muy diferentes entre sí, los modelos de movimiento de objetos no rígidos (por ejemplo, el movimiento humano) y el modelo de movimiento de objetos rígidos (por el ejemplo, el movimiento de vehículos).

El seguimiento basado en modelo del cuerpo humano: El enfoque general para el seguimiento basado en el modelo del cuerpo humano se denomina análisis por síntesis y se conforma por la predicción-comparación-actualización [1], en primer lugar el modelo de la pose de la persona se intenta predecir considerando el conocimiento previo de la escena y la historia del movimiento, a continuación, el modelo es sintetizado y proyectado sobre la trama siguiente y se utiliza una función de evaluación de la pose para medir la similitud entre el modelo proyectado y los datos de la imagen. De acuerdo con diferentes estrategias de búsqueda, esto se hace de forma recursiva, o utilizando técnicas de muestreo hasta que la postura correcta es finalmente encontrada y se utiliza para actualizar el modelo. La estimación de la postura en la primera trama requiere un manejo especial.

De manera general, el algoritmo de seguimiento basados en modelos del cuerpo humano consisten en tres tópicos principales, la construcción de modelos del cuerpo humano, el conocimiento previo de los modelos de movimiento y las propias limitaciones del movimiento humano y finalmente la predicción y estrategias de búsquedas de las posturas generadas durante la forma de caminar del ser humano [20].

La construcción de modelos del cuerpo humano es la base de este método de seguimiento y los modelos pueden ser muy sencillos o muy complejos y el modelo a considerar debe tener en cuenta que modelos mas complejos proporcionan mayor precisión en los resultados de seguimiento, sin embargo esto requiere de un elevado costo computacional. Tradicionalmente, la estructura geométrica del cuerpo humano se puede representar por medio de figuras simples obtenidas por la unión de segmentos de líneas, también puede utilizarse el contorno de las proyecciones en 2-D del cuerpo humano, y generalmente estas utilizan figuras geométricas simples [21]. Sin embargo las proyecciones en 2-D requieren ciertas restricciones del ángulo de visión, es por ello que muchos investigadores utilizan modelos Volumétricos en 3-D usando cuerpos geométricos como esferas, conos, cubos, [48-50] etc. Sin embargo esto presenta un mayor costo computacional. Otros investigadores obtienen buenos resultados usando modelos jerárquicos, es decir, modelan el esqueleto, los músculos con figuras geométricas simples, y modela la piel con superficies poligonales, y una interpretación de las sombras [51].

2.7 DESCRIPCIÓN Y COMPRENSIÓN DEL COMPORTAMIENTO

Después de realizar un exitoso seguimiento del objeto en movimiento de una trama a otra, una nueva etapa del seguimiento de los objetos es la comprensión y descripción del comportamiento de los objetos en movimiento. La comprensión del comportamiento involucra el análisis y reconocimiento de los patrones de movimiento, además de una descripción en lenguaje de alto nivel de las acciones e interacciones de los objetos en movimiento.

a) La Comprensión de los comportamientos sólo puede ser pensada como la clasificación de las características de los datos que varían en el tiempo, es decir, igualando una secuencia de prueba desconocida con unos grupos de secuencias de referencia etiquetadas y que representan comportamientos típicos [59].

Es entonces evidente que un problema fundamental de la comprensión del comportamiento es aprender las secuencias de comportamiento consideradas como referencias y a partir de muestras de entrenamiento, diseñar métodos de entrenamiento y de comparación para enfrentar exitosamente pequeñas variaciones de los datos y de las características dentro de cada clase de los patrones de movimiento.

Uno de los métodos mas populares es la Distorsión Dinámica Temporal (DTW), es un algoritmo para medir la similitud entre dos secuencias que pueden variar en el tiempo o en la velocidad. Por ejemplo, las similitudes en los patrones de la forma de caminar serían detectadas aunque en un video la persona camine lentamente y en otro video la misma persona camina más rápidamente, o incluso si hubo aceleraciones y desaceleraciones en el transcurso de una observación. DTW se ha aplicado a video, audio y gráficos. Por ejemplo, Bobick et al. utilizan DTW para buscar coincidencias de una secuencia de prueba y una secuencia determinista de los estados de ánimo para reconocer los gestos humanos [52, 53].

b) Descripción en lenguaje natural de los comportamientos de los objetos en movimiento.

En muchas aplicaciones de vigilancia visual es importante describir el comportamiento del objeto en un lenguaje natural o de alto nivel que sea apropiado para operadores que no son especialistas en los tópicos de investigación de vigilancia visual [19]. En general, hay dos categorías principales de métodos de descripción del comportamiento: modelos estadísticos y el razonamiento formal. Un modelo estadístico representativo es el modelo de red bayesiana. Este modelo interpreta ciertos eventos y conductas mediante el análisis de secuencias de tiempo y modelos estadísticos [54, 55].

Sin embargo, estos métodos aun trabajan en reconocimiento de conceptos de movimiento de bajo nivel que no involucran conceptos de eventos de alto nivel como la relación entre conceptos de eventos y escenarios [56].

Otro método característico es el razonamiento formalizado que consiste en las características conceptuales de cada segmento como los cambios de posturas y la posición relativa con los otros objetos que lo rodean, con estos datos se seleccionan los verbos y otros elementos sintácticos que construyan un texto en lenguaje natural para interpretar el comportamiento humano que se ha detectado [25, 26].

2.8 IDENTIFICACIÓN DE PERSONAS POR SISTEMAS DE VIGILANCIA VISUAL.

Si el movimiento detectado fue clasificado como el movimiento correspondiente a una persona, el siguiente paso es determinar la identidad de quien esta entrando bajo el área de vigilancia [13], así que de manera conjunta con parámetros biométricos de los seres humanos, tales como: el reconocimiento de rostros, el análisis de la forma de caminar en cada persona lo cual corresponde a un comportamiento especial del movimiento, por esta razón consideramos el análisis de la manera de caminar como una posible solución para la identificación de personas.

Un área de investigación reciente utiliza métodos basados en modelos para medir parámetros como la unión de las trayectorias de movimiento, longitud de extremidades, velocidad angular entre otros

parámetros que permiten la identificación de la manera de caminar muy particular en cada ser humano [28-31].

2.9 FUSIÓN DE DATOS DE MÚLTIPLES CÁMARAS

La detección de movimiento, la comprensión del comportamiento y la identificación de personas que se han discutido previamente, pueden ser realizadas por un sistema de vigilancia visual basado en una sola cámara, sin embargo múltiples cámaras pueden resultar de gran ayuda pues expanden el área de vigilancia [27], además la múltiple información de otras vistas de la misma escena puede ayudar a eliminar el problema de oclusión y manejar un seguimiento continuo [57]. La instalación de múltiples cámaras tiene una gran influencia sobre el desempeño en tiempo real del algoritmo además del costo total del sistema, así es prioritario determinar el número mínimo de cámaras para poder cubrir la escena completa, también es necesario determina los métodos de calibración de las coordenadas 3-D y las coordenadas de la imagen de algunos puntos conocidos para poder calcular los parámetros de la cámara.

Cuando el objeto en movimiento sale del campo de visión de la cámara activa, o la cámara no puede dar una buena visión del objeto en movimiento, el sistema de realizar un cambio a otra cámara que proporcione una mejor visión del objeto.

2.10 SECUENCIAS DE VIDEO EN ESPACIO RGB Y ESCALA DE GRISES.

Una secuencia de video es una secuencia de imágenes ordenadas en el tiempo que representan escenas en movimiento, por lo que un video consta de las dimensiones espaciales de la imagen, es decir el numero de filas y columnas de cada imagen que es constante en toda la secuencia de video, además cada imagen a color es caracterizada en el espacio RGB (también denominada imagen en color verdadero) por tanto cada imagen esta formada por tres planos de color, el rojo (R), el verde (G) y el azul (B). Finalmente cada imagen en el video tiene una ubicación en el tiempo, por tanto una imagen ordena en una secuencia de video la denominaremos trama, la Figura 2.4 muestra el diagrama de una secuencia de video, y los tres planos RGB que componen la trama ocho en color verdadero. Con esta nomenclatura, un pixel determinado por la posición espacial del numero de fila(f) y columna (c), y la posición en el tiempo (t) de la trama que contiene dicho pixel, tiene tres valores de intensidad definidos en el espacio de color RGB, como muestra la ecuación 2.1.

$$p(f, c, t) = \begin{cases} R(f, c, t) \\ G(f, c, t) \\ B(f, c, t) \end{cases} \quad (2.1)$$

Generalmente una secuencia de video esta acompañada de un archivo de audio, sin embargo para este trabajo, no es relevante el audio, por esta razón solo consideramos que la secuencia de video esta compuesta por sus tramas.

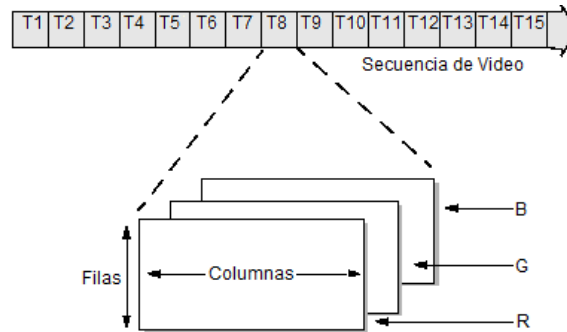


Figura 2. 4 Diagrama de una secuencia de video en espacio RGB

Procesar una imagen en color requiere de mayor tiempo de procesamiento o de mayores recursos de hardware, esto es conocido como costo computacional, dado que se esta considerando que los videos tienen 29.97 tramas por segundo (*fps*) el costo computacional para procesar esta información en el espacio RGB es superior al costo computacional necesario para procesar la imagen en escala de grises, con esta consideración resulta conveniente convertir cada trama del espacio RGB a escala de grises, de este modo la imagen estará definida en un solo arreglo bidimensional, la Figura 2.5, muestra que una imagen en el espacio RGB esta definida en un espacio tridimensional, en tanto que al convertirla en una escala de grises nos permite definirla en un espacio unidimensional finito y discreto, que para una imagen de 8 bits va desde el cero (color negro) hasta el 255 (color blanco).

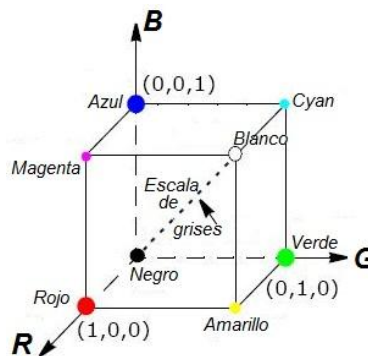


Figura 2. 5 Espacio tridimensional RGB

Para efectuar esta transformación de una imagen en el espacio RGB a una imagen en escala de grises (*ig*), Matlab emplea la función *rgb2gray* que matemáticamente esta definida como:

$$ig = 0.299 \times R + 0.587 \times G + 0.114 \times B \quad (2.2)$$

donde cada *R*, *G*, y *B* son los valores de intensidad de cada plano en la imagen en el espacio RGB.

Por lo que finalmente la secuencia de video estará en escala de grises y por tanto la intensidad de cada pixel en la trama esta definido en su posición de filas (*f*), columnas (*c*) y la ubicación temporal(*t*) de la trama que contiene dicho pixel [63], la Figura 2.6 muestra el diagrama de una secuencia de video en escala de grises.

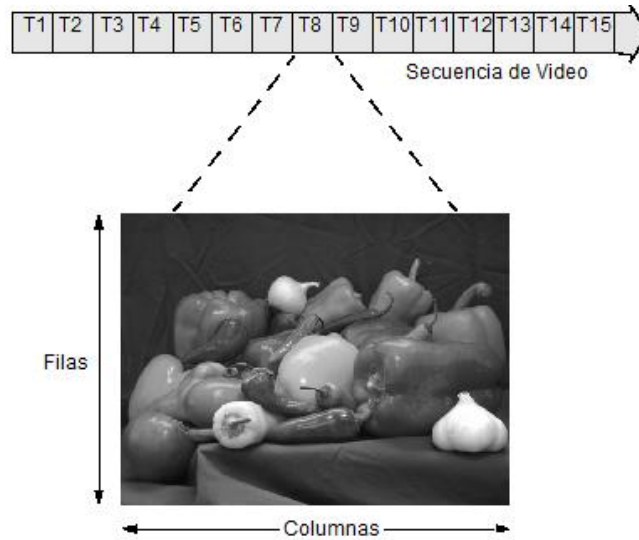


Figura 2. 6 Diagrama de una secuencia de video en escala de grises.

2.11 USO DE LA MAD COMO ESTIMADOR DE LA HOMOGENEIDAD DE LAS REGIONES DE UNA TRAMA

Como se vio en la sección 2.6, un sistema de vigilancia visual puede efectuar la detección de movimiento mediante la segmentación de la imagen en regiones y el análisis de las características de regiones en la trama(t) para compararlas con las características de las regiones de la trama($t + \Delta t$), dado que las regiones de las tramas en escalas de grises tienen valores finitos, discretos y aleatorios, podemos hacer uso de un análisis estadístico para caracterizar el comportamiento de estos datos y determinar si en las tramas que se están analizando existen regiones que tienen cambios significativos que indiquen un movimiento de un objeto en la escena.

Si dividimos la imagen en regiones que a partir de este momento llamaremos ventanas de análisis, proponemos el uso de la Diferencia Absoluta de la Mediana (MAD), un método estadístico robusto para describir el índice de dispersión de los datos que componen la ventana de análisis [61, 62], sea, X un conjunto de k datos, si $x(k)$ es el k –ésimo dato del conjunto de datos, definimos a la MAD como [60]:

$$MAD = MED\{x(k) - MED\{X\}\} \quad (2.3)$$

Donde MED es la mediana del conjunto de datos, la mediana, es el valor que ocupa la posición central en un conjunto de datos ordenados, la ecuación 2.4 se utiliza cuando el número de datos k es impar, en tanto que la ecuación 2.5 se usa cuando el número de datos es par.

$$MED = x_{(k+1)/2} \quad (2.4)$$

$$MED = \frac{x_{k/2} + x_{\frac{k}{2}+1}}{2} \quad (2.5)$$

A la MAD se le conoce como estadística de orden, dado que requiere que los datos que se estén procesando se ordenen ya sea de forma ascendente o descendente, esto le da ventajas significativas en comparación

con estimadores estadísticos clásicos como la varianza o el valor promedio (o media aritmética), la desviación estándar, e incluso la propia mediana. Esto se debe a que si en la muestra existen valores atípicos, es decir, que se encuentran en algún extremo de los posibles valores de la muestra, los estimadores clásicos generaran resultados de baja calidad [60].

Analicemos un conjunto X de $k = 9$ elementos, es decir: $[1,3,2,4,10,4,4,3,2]$ si ordenamos de manera ascendente los elementos del conjunto X tenemos: $[1,2,2,3,3,4,4,4,10]$, ahora dado que $k = 9$, usamos la ecuación 2.4 para calcular la mediana, obteniendo como resultado un 3, mientras que calculando la media aritmética obtenemos como resultado aproximado 3.67 y si realizamos el calculo del valor de la desviación estándar da un resultado aproximado de 2.60. Ahora consideremos que el valor de 10 en el conjunto de datos es cambiado por 20 $[1,2,2,3,3,4,4,4,20]$ entonces los valores calculados son, mediana = 3, media = 4.78, y la desviación estándar = 5.81.

Esto demuestra que la estadística de orden mantiene un valor de tendencia central, en tanto que los otros estimadores usados muestran datos poco confiables. Ahora mostraremos en la tabla 2.1 el concepto del procedimiento usado para calcular la MAD.

DATOS	DATOS ORDENADOS	DIFERENCIA
1	1	-2
3	2	-1
2	2	-1
4	3	0
10	3	0
4	4	1
4	4	1
3	4	1
2	10	7

Tabla 2. 1 Operaciones efectuadas para el cálculo de la MAD

En la primera columna tenemos el conjunto de datos, en la segunda columna se presentan los datos ordenados, y dado que la mediana calculada para estos valores es 3, la tercer columna muestra los resultados obtenidos de restar la mediana a cada elemento del conjunto de datos ordenados, y finalmente de esta columna se calcula nuevamente la mediana que ahora será el valor de la MAD y que en este caso es cero. La Tabla 2.2 muestra que si realizamos el cambio del elemento de valor 10, por un elemento de valor 20 los resultados serán los mismos.

Con estas consideraciones proponemos el uso de la MAD como un estimador estadístico robusto y cuyo significado físico en el procesamiento de imágenes lo consideramos como un índice de homogeneidad de los

datos, así si los datos son homogéneos, es decir los pixeles son del mismo color, el valor calculado de la MAD será cero, en tanto que si los colores en los pixeles están dispersos o en esa ventana de análisis existen regiones que delimitan algún objeto, la MAD será distinta de cero.

DATOS	DATOS ORDENADOS	DIFERENCIA
1	1	-2
3	2	-1
2	2	-1
4	3	0
20	3	0
4	4	1
4	4	1
3	4	1
2	20	17

Tabla 2. 2 Operaciones efectuadas para el cálculo de la MAD, mostrando la inserción de un dato atípico

Considerando entonces que la MAD es un estimador estadístico robusto, proponemos el desarrollo de un algoritmo que separe cada trama en ventanas de análisis y calcule la MAD en cada una de las ventanas de análisis, posteriormente haremos una comparación entre los resultados obtenidos con las ventanas de la trama(t) y las correspondientes de la trama(t + Δt) para determinar si en esa ventana de análisis ocurrió un cambio importante en la escena que indique que ahí existe un movimiento significativo.

2.12 CONCLUSIONES

Los sistemas de vigilancia visual ofrece soluciones viables a las crecientes necesidades de la sociedad por sistemas que mejoren la calidad de vida en temas como la seguridad por ejemplo en situaciones donde es necesario alertar al personal de seguridad si un objeto esta entrando o saliendo de un área vigilada, si ese objeto es una persona o un vehículo, y posteriormente indicar si este vehículo o persona se encuentra autorizado. También propone soluciones en situaciones donde se requiere contabilizar el número de personas o vehículos que entran o salen de un área determinada, o si el comportamiento que presentan es anormal indicando un accidente, o alguna actividad ilícita. Para obtener estos resultados finales el sistema de vigilancia ejecuta etapas subsecuentes a la detección de movimiento, por ello realizamos una propuesta basada en el estimador estadístico robusto de la MAD para la detección de movimiento basados en el concepto de comparar las características de una región en la trama actual con las características de la misma región de la trama siguiente.

2.13 REFERENCIAS BIBLIOGRÁFICAS

- [1] H. Weiming, T. Tieniu, W. Liang and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 34(3), 334-352 (2004).
- [2] I. Ahmad, Z. He, M. Liao, F. Pereira and M. T. Sun, "Special Issue on Video Surveillance," *Circuits and Systems for Video Technology, IEEE Transactions on* 18(8), 1001-1005 (2008).
- [3] X. Li-Qun, "Issues in video analytics and surveillance systems: Research / prototyping vs. applications / user requirements," in *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pp. 10-14 (2007).
- [4] R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, and L. Wixson, "A system for video surveillance and monitoring," Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep., CMU-RI-TR-00-12, 2000.
- [5] I. Haritaoglu, D. Harwood, and L. S. Davis, "W : Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 809–830, Aug. 2000.
- [6] A. Baumberg and D. C. Hogg, "Learning deformable models for tracking the human body," in *Motion-Based Recognition*, M. Shah and R. Jain, Eds. Norwell, MA: Kluwer, 1996, pp. 39–60.
- [7] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 780–785, July 1997.
- [8] A. J. Lipton, H. Fujiyoshi, and R. S. Patil, "Moving target classification and tracking from real-time video," in *Proc. IEEE Workshop Applications of Computer Vision*, 1998, pp. 8–14.
- [9] S. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking groups of people," *Comput. Vis. Image Understanding*, vol. 80, no. 1, pp. 42–56, 2000.
- [10] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, 1999, pp. 246–252.
- [11] D. Meyer, J. Denzler, and H. Niemann, "Model based extraction of articulated objects in image sequences for gait analysis," in *Proc. IEEE Int. Conf. Image Processing*, 1998, pp. 78–81.
- [12] J. Barron, D. Fleet, and S. Beauchemin, "Performance of optical flow techniques," *Int. J. Comput. Vis.*, vol. 12, no. 1, pp. 42–77, 1994.
- [13] N. Friedman and S. Russell, "Image segmentation in video sequences: a probabilistic approach," in *Proc. 13th Conf. Uncertainty in Artificial Intelligence*, 1997, pp. 1–3.
- [14] E. Stringa, "Morphological change detection algorithms for surveillance applications," in *Proc. British Machine Vision Conf.*, 2000, pp. 402–412.
- [15] Y. Kuno, T. Watanabe, Y. Shimosakoda, and S. Nakagawa, "Automated detection of human for visual surveillance system," in *Proc. Int. Conf. Pattern Recognition*, 1996, pp. 865–869.
- [16] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection, analysis, and applications," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 781–796, Aug. 2000.
- [17] A. J. Lipton, "Local application of optic flow to analyze rigid versus nonrigid motion," in *Proc. Int. Conf. Computer Vision Workshop Frame-Rate Vision*, Corfu, Greece, 1999.
- [18] D. Meyer, J. Psl, and H. Niemann, "Gait classification with HMM's for trajectories of body parts extracted by mixture densities," in *Proc. British Machine Vision Conf.*, 1998, pp. 459–468.

- [19] J. G. Lou, Q. F. Liu, W. M. Hu, and T. N. Tan, "Semantic interpretation of object activities in a surveillance system," in *Proc. Int. Conf. Pattern Recognition*, 2002, pp. 777–780.
- [20] J. K. Aggarwal, Q. Cai, W. Liao, and B. Sabata, "Non-rigid motion analysis: articulated & elastic motion," *Comput. Vis. Image Understanding*, vol. 70, no. 2, pp. 142–156, 1998.
- [21] S. Ju, M. Black, and Y. Yacobb, "Cardboard people: a parameterized model of articulated image motion," in *Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition*, 1996, pp. 38–44.
- [22] N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 266–280, Mar. 2000.
- [23] R. Polana and R. Nelson, "Low level recognition of human motion," in *Proc. IEEE Workshop Motion of Non-Rigid and Articulated Objects*, Austin, TX, 1994, pp. 77–82.
- [24] D.-S. Jang and H.-I. Choi, "Active models for tracking moving objects," *Pattern Recognit.*, vol. 33, no. 7, pp. 1135–1146, 2000.
- [25] M. Izumi and A. Kojima, "Generating natural language description of human behaviors from video images," in *Proc. Int. Conf. Pattern Recognition*, 2000, pp. 728–731.
- [26] A. Kojima, T. Tamura, and K. Fukunaga, "Natural language description of human activities from video images based on concept hierarchy of actions," *Int. J. Comput. Vis.*, vol. 50, no. 2, pp. 171–184, 2002.
- [27] Q. Cai and J. K. Aggarwal, "Tracking human motion using multiple cameras," in *Proc. Int. Conf. Pattern Recognition*, Vienna, Austria, 1996, pp. 68–72.
- [28] S. A. Niyogi and E. H. Adelson, "Analyzing gait with spatio-temporal surface," in *Proc. IEEE Workshop Motion of Non-Rigid and Articulated Objects*, 1994, pp. 64–69.
- [29] C. BenAbdelkader, R. Cutler, H. Nanda, and L. Davis, "EigenGait: motion- based recognition of people using image self-similarity," in *Proc. Int. Conf. Audio- and Video-Based Biometric Person Authentication*, 2001, pp. 312–317.
- [30] L. Lee and W. Grimson, "Gait analysis for recognition and classification," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Washington, DC, 2002, pp. 155–162.
- [31] A. Kale, A. Rajagopalan, N. Cuntoor, and V. Kruger, "Gait-based recognition of humans using continuous HMMs," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Washington, DC, 2002, pp. 336–341.
- [32] M. Köhle, D. Merkl, and J. Kastner, "Clinical gait analysis by neural networks: Issues and experiences," in *Proc. IEEE Symp. Computer-Based Medical Systems*, 1997, pp. 138–143.
- [33] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE Trans. Pattern Recognit. Machine Intell.*, vol. 23, pp. 349–361, Apr. 2001.
- [34] A. Galata, N. Johnson, and D. Hogg, "Learning variable-length Markov models of behavior," *Comput. Vis. Image Understanding*, vol. 81, no. 3, pp. 398–413, 2001.
- [35] Y. Wu and T. S. Huang, "A co-inference approach to robust visual tracking," in *Proc. Int. Conf. Computer Vision*, vol. II, 2001, pp. 26–33.
- [36] H. Z. Sun, T. Feng, and T. N. Tan, "Robust extraction of moving objects from image sequences," in *Proc. Asian Conf. Computer Vision*, Taiwan, R.O.C., 2000, pp. 961–964.

- [37] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using adaptive tracking to classify and monitor activities in a site," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Santa Barbara, CA, 1998, pp. 22–31.
- [38] C. Ridder, O. Munkelt, and H. Kirchner, "Adaptive background estimation and foreground detection using Kalman-filtering," in *Proc. Int. Conf. Recent Advances in Mechatronics*, 1995, pp. 193–199.
- [39] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russel, "Toward robust automatic traffic scene analysis in real-time," in *Proc. Int. Conf. Pattern Recognition*, Israel, 1994, pp. 126–131.
- [40] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: principles and practice of background maintenance," in *Proc. Int. Conf. Computer Vision*, 1999, pp. 255–261.
- [41] K. Karmann and A. Brandt, "Moving object recognition using an adaptive background memory," in *Time-Varying Image Processing and Moving Object Recognition*, V. Cappellini, Ed. Amsterdam, The Netherlands: Elsevier, 1990, vol. 2.
- [42] M. Kilger, "A shadow handler in a video-based real-time traffic monitoring system," in *Proc. IEEE Workshop Applications of Computer Vision*, Palm Springs, CA, 1992, pp. 11–18.
- [43] T. J. Fan, G. Medioni, and G. Nevatia, "Recognizing 3-D objects using surface descriptions," *IEEE Trans. Pattern Recognit. Machine Intell.*, vol. 11, pp. 1140–1157, Nov. 1989.
- [44] B. Coifman, D. Beymer, P. McLauchlan, and J. Malik, "Areal-time computer vision system for vehicle tracking and traffic surveillance," *Transportation Res.: Part C*, vol. 6, no. 4, pp. 271–288, 1998.
- [45] J. Malik and S. Russell, "Traffic surveillance and detection technology development (new traffic sensor technology)," Univ. of California, Berkeley, 1996.
- [46] C. A. Pau and A. Barber, "Traffic sensor using a color vision method," in *Proc. SPIE—Transportation Sensors and Controls: Collision Avoidance, Traffic Management, and ITS*, vol. 2902, 1996, pp. 156–165.
- [47] B. Schiele, "Voxel-free tracking of cars and people based on color regions," in *Proc. IEEE Int. Workshop Performance Evaluation of Tracking and Surveillance*, Grenoble, France, 2000, pp. 61–71.
- [48] Q. Delamarre and O. Faugeras, "3D articulated models and multi-view tracking with physical forces," *Comput. Vis. Image Understanding*, vol. 81, no. 3, pp. 328–357, 2001.
- [49] , "3D articulated models and multi-view tracking with silhouettes," in *Proc. Int. Conf. Computer Vision*, Kerkyra, Greece, 1999, pp. 716–721.
- [50] C. Sminchisescu and B. Triggs, "Covariance scaled sampling for monocular 3D body tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Kauai, HI, 2001, pp. 1:447–1:454.
- [51] R. Plankers and P. Fua, "Articulated soft objects for video-based body modeling," in *Proc. Int. Conf. Computer Vision*, Vancouver, BC, Canada, 2001, pp. 394–401.
- [52] K. Takahashi, S. Seki, H. Kojima, and R. Oka, "Recognition of dexterous manipulations from time varying images," in *Proc. IEEE Workshop Motion of Non-Rigid and Articulated Objects*, Austin, TX, 1994, pp. 23–28.
- [53] A. F. Bobick and A. D. Wilson, "A state-based technique to the representation and recognition of gesture," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 1325–1337, Dec. 1997.

- [54] T. Huang, D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell, and J. Weber, "Automatic symbolic traffic scene analysis using belief networks," in *Proc. National Conf. Artificial Intelligence*, 1994, pp. 966–972.
- [55] P. Remagnino, T. Tan, and K. Baker, "Agent orientated annotation in model based visual surveillance," in *Proc. IEEE Int. Conf. Computer Vision*, 1998, pp. 857–862.
- [56] M. Mohnhaupy and B. Neumann, "On the use of motion concepts for top-down control in traffic scene," in *Proc. European Conf. Computer Vision*, 1990, pp. 542–550.
- [57] S. L. Dockstader and A. M. Tekalp, "Multiple camera tracking of interacting and occluded human motion," *Proc. IEEE*, vol. 89, pp. 1441–1455, Oct. 2001.
- [58] O. Javed and M. Shah, "Tracking and object classification for automated surveillance," in *Proc. European Conf. Computer Vision*, vol. 4, 2002, pp. 343–357.
- [59] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Trans. Pattern Recognit. Machine Intell.*, vol. 23, pp. 257–267, Mar. 2001.
- [60] T. Pham-Gia and T. L. Hung, "The mean and median absolute deviations," *Mathematical and Computer Modelling* 34(7–8), 921-936 (2001).
- [61] F. J. Gallegos-Funes, J. Martinez-Valdes, J. M. De-la-Rosa-Vázquez, "Order statics filters in wavelet domain for color image processing" *IEEE Sixth Mexican Int. Conf. on Artificial Intelligence, MICA12007*, pp. 121-130, México, 2007.
- [62] F. J. Gallegos-Funes, V. I. Ponomaryov, "Real-time image filtering scheme based on robust estimators in presence of impulsive noise" *Real Time Imaging*, vol.10, No. 2, pp. 6980, 2004.
- [63] "R2012a Documentation, Image Processing Toolbox, Morphology Fundamentals, Working with Image Data in the MATLAB Workspace" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/imaq/f12-139775.html>

3.1 INTRODUCCIÓN

En este capítulo se presentan los métodos y algoritmos propuestos para detectar el movimiento de objetos en secuencias de video, la aplicación del algoritmo y los resultados mostrados en cada etapa permiten realizar evoluciones y mejoras pues cada análisis muestra las necesidades a considerar para el desarrollo de un algoritmo más robusto.

3.2 **ALGORITMO A** PARA SECUENCIAS DE VIDEO SIMULADAS

El uso de la *MAD* (capítulo 2, sección 11) para la detección de objetos en movimiento en secuencias de video se confirma en esta sección, para tal efecto se realiza una secuencia de video simulada, es decir, una imagen se repite en cada trama de la secuencia de video y en cada trama se inserta un objeto que es una sección de otra imagen cuyas intensidades de los píxeles difieren respecto a la trama base. La secuencia de video se realiza insertando un objeto de manera que de la apariencia de movimiento continuo tanto en dirección horizontal, vertical y diagonal. La construcción de esta secuencia de video nos permite la aplicación del **Algoritmo A**, la Figura 3.1 describe su procedimiento.

Considerando el diagrama de flujo de la Figura 3.1, y que la secuencia de video simulada ha sido construida previamente (para la realización de las primeras pruebas no se utiliza el formato QCIF) y teniendo la secuencia de imágenes ordenadas en el tiempo, se procede a tomar las tramas de la secuencia de video simulada.

Las imágenes o tramas presentadas en el espacio de color RGB requieren de un mayor costo computacional y un mayor tiempo de procesamiento, por lo que se considera realizar la conversión de imágenes del espacio de color RGB a imágenes en escala de grises, de este modo se optimiza el costo computacional y el tiempo de procesamiento (véase la sección 2.10).

Cada trama se ordena en el espacio temporal, es decir, se respeta el orden de la secuencia de video simulada, dicha secuencia se compone de tramas que están definidas por el número de filas, columnas y la escala temporal (f, c, t) [31]. La Figura 3.2 ilustra este procedimiento, además muestra que la imagen es subdividida en ventanas de análisis de $m \times n$ píxeles y que para fines demostrativos están delimitadas por líneas en color verde, las ventanas de análisis de la *Trama1* se organizan en un arreglo matricial A_{ij} , y es identificada como la trama actual o la trama obtenida en el instante t . La *Trama2* es la trama anterior o a la trama obtenida en el instante $t-1$, cuyo arreglo matricial de las ventanas de análisis se denomina B_{ij} . A cada una de las ventanas de análisis se calcula la *MAD*, obteniendo los arreglos matriciales $MAD_{A_{ij}}$ y $MAD_{B_{ij}}$ para cada trama procesada.

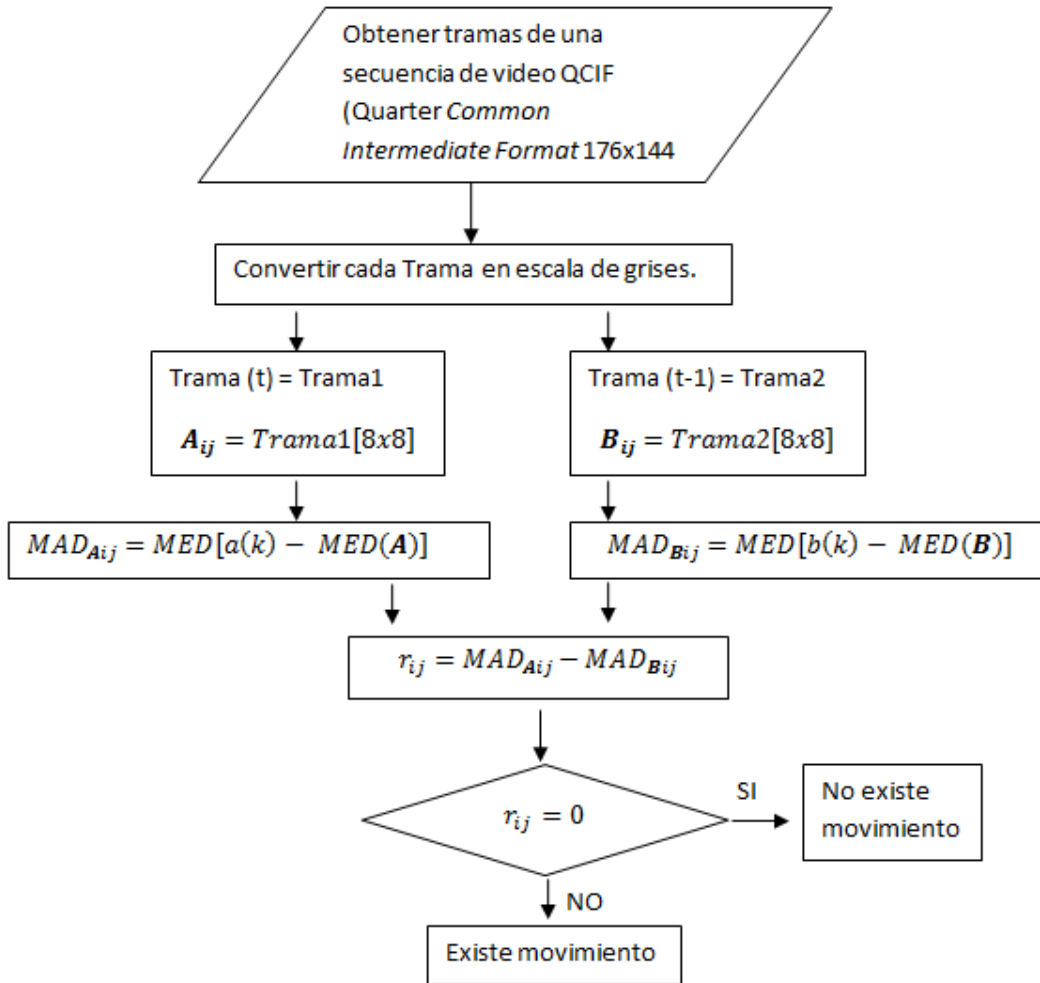


Figura 3. 1 Diagrama de flujo del Algoritmo A.

Finalmente, se restan los resultados correspondientes de los arreglos matriciales MAD_{Aij} y MAD_{Bij} almacenándose en el arreglo r_{ij} [28-30]. En la Figura 3.2 también se muestra una marca que indica una ventana de análisis en la trama actual y su correspondiente en la trama anterior, la cual también es usada para realizar el criterio de movimiento, es decir, si el valor del elemento de la matriz r_{ij} es distinto de cero indica que en esa ventana de análisis existe un cambio y por tanto un movimiento, los valores de r_{ij} pueden ser positivos y negativos. En adelante se considera que los índices i, j se originan a partir de la posición de las ventanas de análisis, y por tanto los arreglos matriciales subsecuentes tienen los mismos índices.

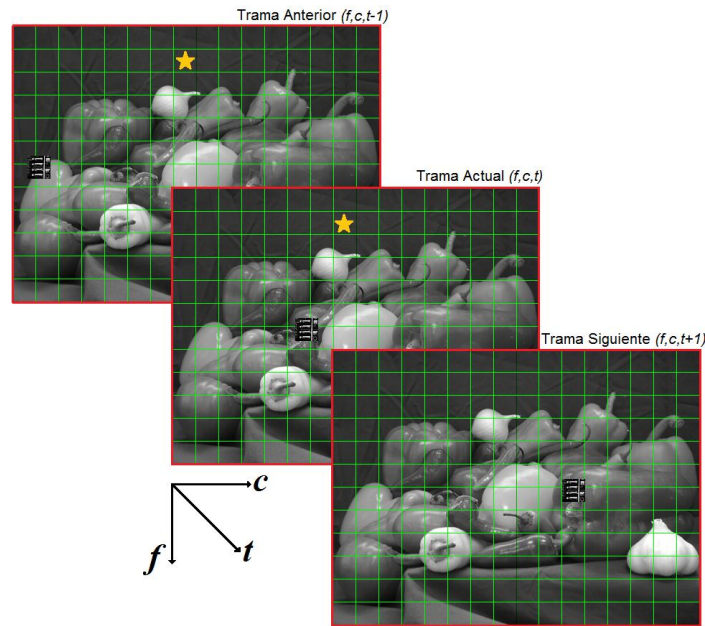


Figura 3. 2 Secuencia de video simulada con un objeto en movimiento en dirección horizontal.

Los lugares de la ventana de análisis donde las intensidades de los pixeles permanecen sin cambio entre la trama actual y la trama anterior, dado que el objeto que simula el movimiento continuo no fue insertado en esa posición de la ventana de análisis, nos proporciona el mismo valor de la *MAD*, y al restar las *MAD* obtenidas de las ventanas de análisis correspondientes dan como resultado cero, mientras que las ventanas de análisis donde el objeto insertado ingresa o sale de la ventana de análisis nos da un valor distinto de cero y además los valores obtenidos van desplazándose y cambiando el signo conforme el objeto insertado entra o sale de la ventana de análisis. Los resultados obtenidos en los movimientos simulados en direcciones: vertical, horizontal y diagonal presentan el mismo comportamiento.

3.3 ALGORITMO B: IMPLEMENTACION DE UN UMBRAL DINAMICO PARA DISCRIMINAR MOVIMIENTO NO SIGNIFICATIVO EN SECUENCIAS DE VIDEO REALES.

Las secuencias de video simuladas presentan un caso ideal para la detección de movimiento, pues los valores de la imagen de fondo no presentan cambio alguno y por ello la detección de movimiento puede realizarse con el **Algoritmo A**, sin embargo, en escenas reales existen cambios que son producto de diversos factores como iluminación, movimientos no significativos [10] y sombras, por esta razón, se ve la necesidad de implementar un umbral que permita discriminar los movimientos no significativos.

El **Algoritmo B** mostrado en la Figura 3.3 es una mejora del **Algoritmo A** detallado en la sección 3.1. En tal algoritmo se determinó que el criterio de movimiento se basa en el valor de la matriz r_{ij} si el valor es distinto de cero existe movimiento, pero ahora vemos la necesidad de realizar un umbral para discriminar el movimiento significativo, el umbral se calcula con la hipótesis planteada a continuación.

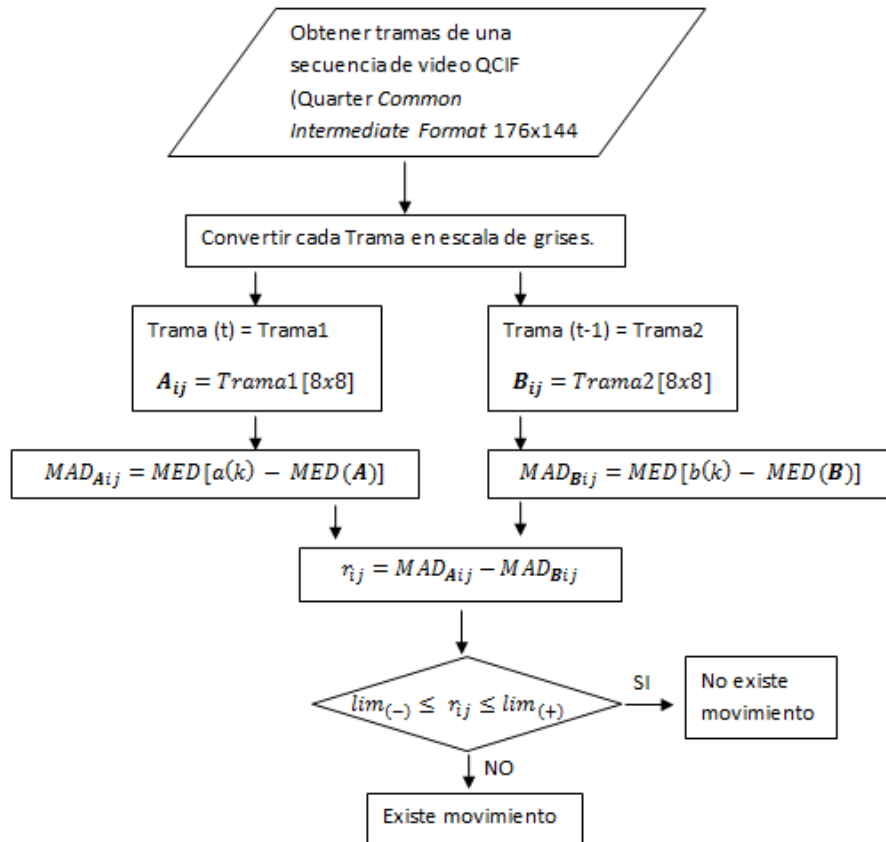


Figura 3. 3 Diagrama de flujo del **Algoritmo B**.

Cuando el valor de la matriz r_{ij} es cero indica que en la ventana de análisis $MAD_{Aij} = MAD_{Bij}$, sin embargo, cuando existen cambios por factores como la iluminación en las escenas ó el objeto en movimiento ocupa un pequeño espacio en la ventana de análisis, el valor del elemento del arreglo r_{ij} es cercano a cero, entonces el cambio entre las tramas es pequeño. Además, cuando existe un cambio de iluminación que afecta a toda la escena la mayor parte de los elementos del arreglo r_{ij} son distintos de cero. Con este argumento es difícil tener un valor fijo como umbral, pues dependiendo si en una escena existe mayor o menor iluminación los valores serán distintos en signo y valor, por tal motivo es necesario considerar un umbral dinámico para cada par de escenas procesadas [8].

En la Figura 3.4 se muestra un ejemplo de esta particularidad , pues en ventanas de análisis donde no existe movimiento significativo, el **Algoritmo A** determinó un movimiento significativo, y como

se explico anteriormente, en esta secuencia de imágenes es probable que tal resultado sea producto de un cambio de iluminación en alguna de las escenas. Considerando que las tramas fueron divididas en ventanas de análisis los resultados de las operaciones efectuadas sobre tales ventanas de análisis se almacenan en los diversos arreglos designados (MAD_{Aij} , MAD_{Bij} , y r_{ij}), para que los resultados obtenidos con los algoritmos nos presenten una clara información visual, se realizó una modificación del contraste, cuando el resultado del **Algoritmo A** fue igual a uno se realizó un incremento en el contraste de la ventana de análisis que genero ese resultado, en tanto que si se obtuvo un cero, el contraste se disminuyó haciendo que en esa ventana la imagen se viera transparente [1-3][5].

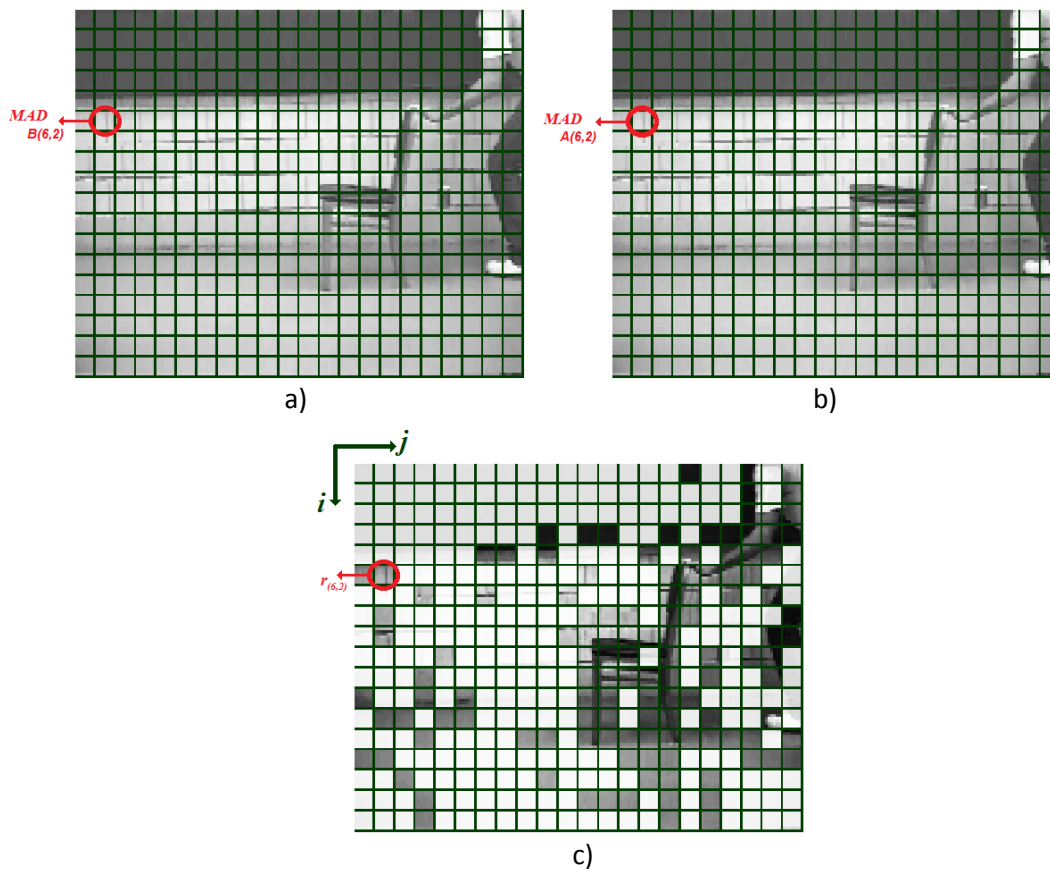


Figura 3. 4 a) Trama anterior: trama 10 en la secuencia de video “Chair”, b) Trama Actual: trama 11 en la secuencia de video “Chair”, c) Modificación del contraste en las ventanas de análisis basados en el arreglo r_{ij} del **Algoritmo A**

Para ejemplificar las operaciones y resultados obtenidos nos enfocaremos en la ventana de análisis $r_{(6,2)}$ en donde podemos apreciar que visualmente no existe diferencia, y que no existe un objeto moviéndose en tal ventana de análisis, pero que el **Algoritmo A** indica que en esa ventana de análisis hay movimiento.

En la Figura 3.5 se muestran las ventanas de análisis con los valores obtenidos desde la trama actual y considerando una ventana de análisis que contenga los pixeles desde la fila 41 hasta la 48 y de la columna 9 hasta la 16 es decir: $Trama(11)_{(41:48,9:16)}$ y la trama anterior con la misma característica de la ventana de análisis, es decir: $Trama(10)_{(41:48,9:16)}$ en donde los cálculos correspondientes son $MAD_{A(6,2)} = 9$ y $MAD_{B(6,2)} = 8.5$, [4], [6], [7], [16], [17], [18] en tanto que el resultado de la ventana de análisis de interés es $r_{(6,2)} = -0.5$. Dado que el **Algoritmo A** solo considera dos opciones, si el resultado en esa ventana de análisis fuera igual a cero indicaría que no existe movimiento sin embargo si el resultado fuera diferente de cero, como en el caso particular que estamos tratando, entonces se considera que la ventana de análisis presenta movimiento, pero como se muestra a continuación no existe un cambio significativo entre los valores de las ventanas de análisis de las tramas analizadas.

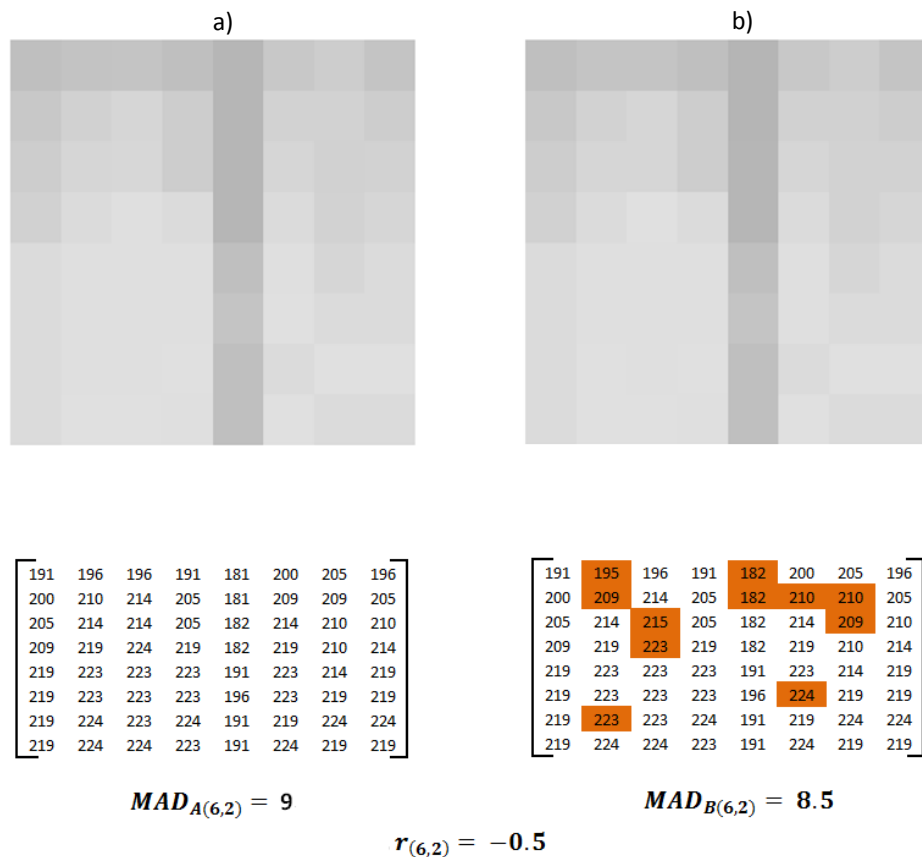


Figura 3.5 Ventanas de análisis (6,2) a) Tramas 11 (Actual), b) 10 (Anterior) con los valores y el nivel de gris que representan, el resultado del cálculo de la $MAD_{A(6,2)}$, la $MAD_{B(6,2)}$ y el resultado $r_{(6,2)}$

Con estos resultados se vió la necesidad de que el umbral debe tener la capacidad de contemplar valores positivos y negativos además de ser adaptivo a las tramas que se están analizando, por lo que se propuso analizar el histograma [4], [6], [7] del arreglo r_{ij} . El histograma se realiza calculando el valor mínimo y máximo del arreglo r_{ij} . Consideramos estos valores para formar el eje horizontal que representa las marcas de clase (bin) o intervalos en los que están agrupados los datos, para nuestro análisis, tales intervalos tienen una unidad de separación, el eje vertical muestra la frecuencia de ocurrencia del valor calculado con la resta de las MAD.

Con la ayuda de la representación gráfica mostrada en la Figura 3.6 observamos que la mayoría de los resultados se encuentran concentrados alrededor del cero ó en valores cercanos al cero, esto es, porque cuando no existe un movimiento significativo los valores son cero o muy cercanos a cero aproximándose desde la izquierda (valores negativos) o desde la derecha (valores positivos), con esta consideración podemos afirmar que el umbral puede determinarse dinámicamente con los resultados obtenidos entre cada trama [4], [6], [7].

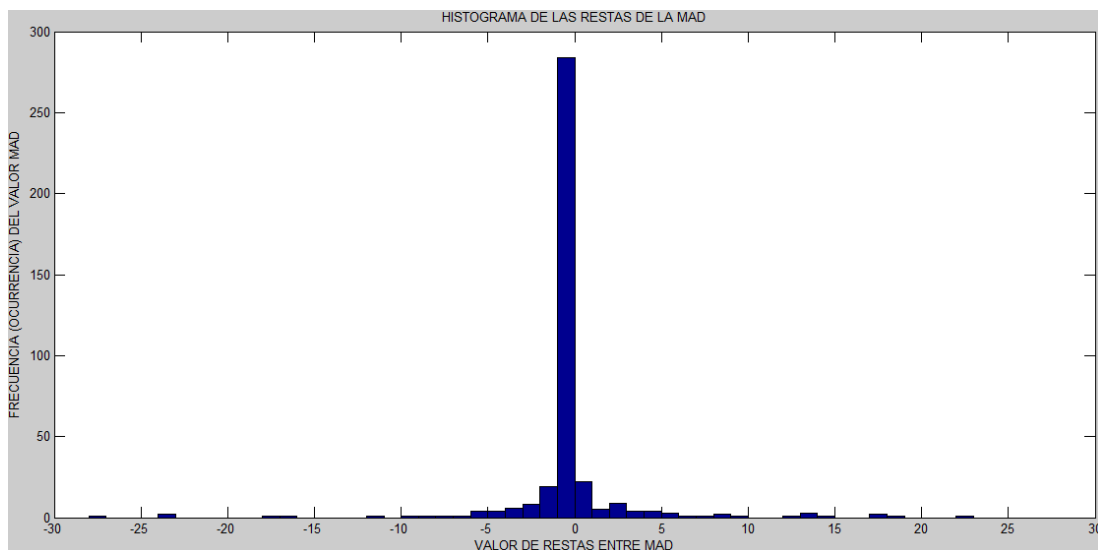


Figura 3. 6 Histograma de las restas de las MAD para las Trama 10 y 11 del video “Chair”

Ahora, dado que los valores obtenidos pueden modelarse con una función de distribución de probabilidad normal o Gaussiana [4], [6], [7], inferimos que los límites pueden ser determinados al calcular el valor promedio o media aritmética \bar{r} y el valor de la desviación estándar r_{σ} , para realizar estos cálculos consideramos los valores de r_{ij} como un vector con k elementos es decir $k = i \times j$ y denominamos a d como el índice de cada elemento dicho vector, por lo que las formulas utilizadas para efectuar dichos cálculos y la ecuación para modelar la función de distribución Gaussiana son:

$$\bar{r} = \frac{1}{k} \sum_{e=1}^k r_d \quad (3.1)$$

$$r_\sigma = \left[\frac{\sum_{d=1}^k (\bar{r} - r_\sigma)^2}{k-1} \right]^{\frac{1}{2}} \quad (3.2)$$

$$P(r) = \frac{1}{\sqrt{2\pi}r_\sigma} e^{-\frac{1}{2} \left[\frac{\bar{r}-r}{r_\sigma} \right]^2} \quad (3.3)$$

Con estos argumentos se propone que el umbral este determinado por dos límites, el límite negativo definido como $lim_{(-)} = \bar{r} - r_\sigma$ y el límite positivo presentado como $lim_{(+)} = \bar{r} + r_\sigma$. Para el caso de las Tramas 10 y 11 que se están analizando los resultados obtenidos son: $lim_{(-)} = \bar{r} - r_\sigma = 0.1073 - 3.9651 = -3.8578 \approx -3.9$ y $lim_{(+)} = \bar{r} + r_\sigma = 0.1073 + 3.9651 = 4.0724 \approx 4.1$. Con los valores r_{ij} obtenidos de las restas de las MAD se realiza la discriminación del movimiento significativo considerando que si $lim_{(-)} \leq r_{ij} \leq lim_{(+)}$ entonces no existe movimiento significativo, en caso contrario se considera como movimiento significativo, esta es la mejora del **Algoritmo B** respecto al **Algoritmo A**. La Figura 3.7 muestra el modelado de la distribución normal de las restas de las MAD y los valores obtenidos para \bar{r} , r_σ , $lim_{(-)}$ y $lim_{(+)}$ obtenidos en las Tramas 11 y 10 de la secuencia de Video "Chair".

Retomando la ventana de análisis $r_{(6,2)} = -0.5$, su resultado esta dentro del intervalo considerado como análisis no significativo, y por lo tanto esta ventana de análisis no es considerada movimiento significativo.

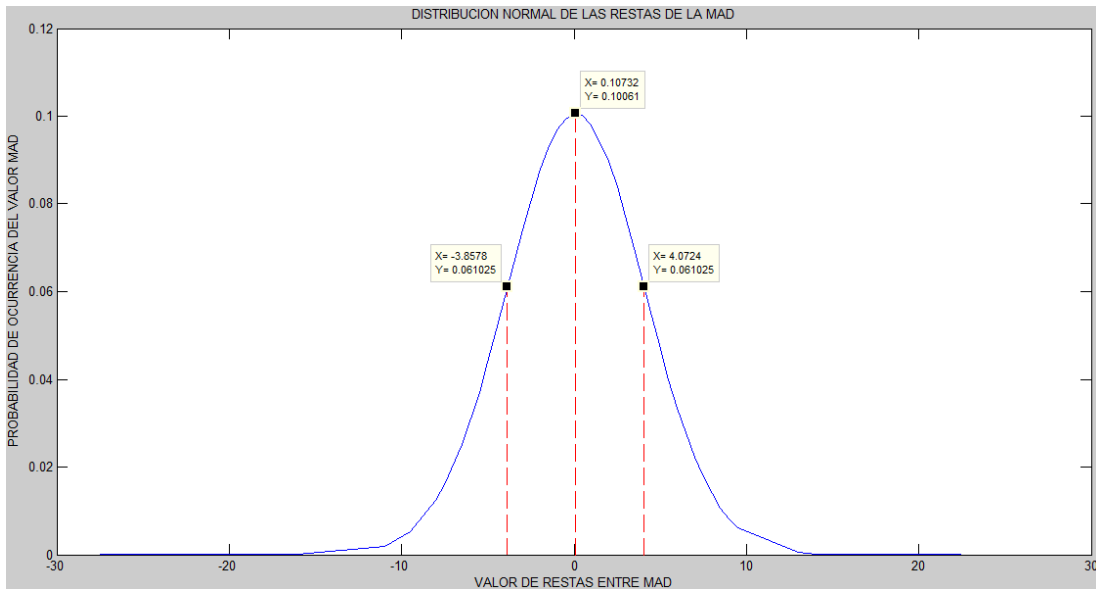


Figura 3. 7 Distribución Normal de las restas de las MAD y los valores obtenidos para los limites positivo y negativo en las Tramas 10 y 11 del video "Chair"

La Figura 3.8 muestra un comparativo de los resultados obtenidos con el **Algoritmo A** y el **Algoritmo B** aplicados en las tramas 10 y 11 de la secuencia de video "Chair". Nuevamente para tener una mejor visualización de los resultados obtenidos se realiza una mejora del contraste [1-3], [5] a las ventanas en donde se ha considerado que existe movimiento, en tanto que las ventanas en las que no presentaron movimiento se realiza una modificación del contraste para hacer transparentes esas ventanas de análisis [9].

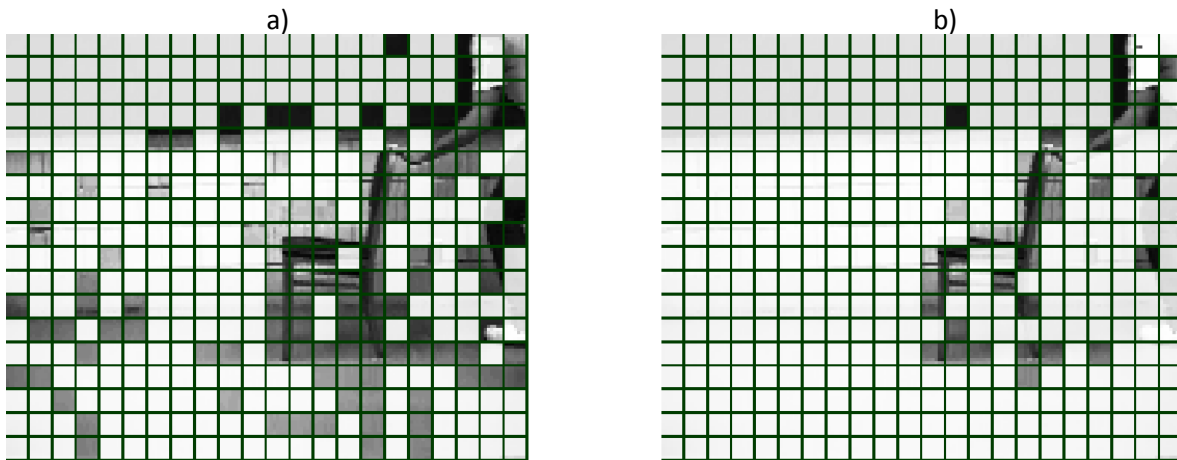


Figura 3. 8 Resultados obtenidos con la aplicación a) Algoritmo A y b) Algoritmo B a las Tramas 11 y 10 de la secuencia de video "Chair".

3.4 IMPLEMENTACION DE TECNICAS DE PREPROCESAMIENTO EN LAS TRAMAS Y EL CAMBIO DEL TAMAÑO EN LAS VENTANAS DE ANALISIS.

La implementación de técnicas de pre-procesamiento o mejoramiento de la imagen, tienen como objetivo realizar un filtrado o una transformación que mejore o realce determinadas características de la imagen para una mejor apreciación visual o mejorar el desempeño del algoritmo o procesamiento principal de un sistema de visión artificial. Estas tareas pueden ser efectuadas en el dominio de la frecuencia usando la Transformada de Fourier de la imagen ó en el dominio espacial [13], el termino dominio espacial se refiere al conjunto de puntos que componen una imagen y los métodos en el dominio espacial son procedimientos que operan directamente sobre los pixeles. En el dominio espacial también es posible efectuar operaciones que comúnmente se efectúan en el dominio de la frecuencia, las transformaciones de procesamiento de imágenes en el dominio espacial se pueden expresar como [1], [3],

$$\beta(f, c) = \Gamma[\alpha(f, c)] \tag{3.4}$$

donde $\alpha(f, c)$ es la imagen de entrada, $\beta(f, c)$ es la imagen procesada y Γ es un operador que actúa sobre $\alpha(f, c)$ definida en alguna vecindad del punto (f, c) , cuando la vecindad tiene un tamaño de 1x1 pixel la transformación se conoce como función de transformación de intensidad y no depende del entorno del pixel a procesar y se puede expresar como [1], [3],

$$s = \Gamma(\zeta) \quad (3.5)$$

donde s y ζ con variables que denotan respectivamente la intensidad de la imagen de salida y de entrada en cualquier punto (f, c) .

La detección de movimiento mediante el uso de la *MAD* [16], [17], [18] es sensible a pequeñas variaciones, como se aprecia en la Figura 3.5, por lo que se propone eliminar cierta información de los pixeles mediante el uso de una función de transformación de intensidad para que este método no sea alterado por la vecindad del pixel.

Las Tramas que se están procesando se han transformado a niveles de grises (ver **Algoritmo A**) y estos niveles de grises se representan en un byte, lo que implica que cada pixel de la imagen tiene 8 bits para formar el nivel de intensidad de ese pixel [12], proponemos realizar un recorte del plano de bits de la imagen para conocer la contribución de cada plano de bits en la imagen [10]. Supongamos que la imagen se compone de 8 planos de 1 bit, desde el plano de bits 0 (el bit menos significativo, LSB), hasta el plano de bits 7 (más significativo MSB). El plano de bits cero contiene todos los LSB de cada byte de cada pixel en la imagen y el plano 7 [1], [2], [3] contiene los MSB de cada byte de cada pixel de la imagen, este concepto se representa en la Figura 3.9.

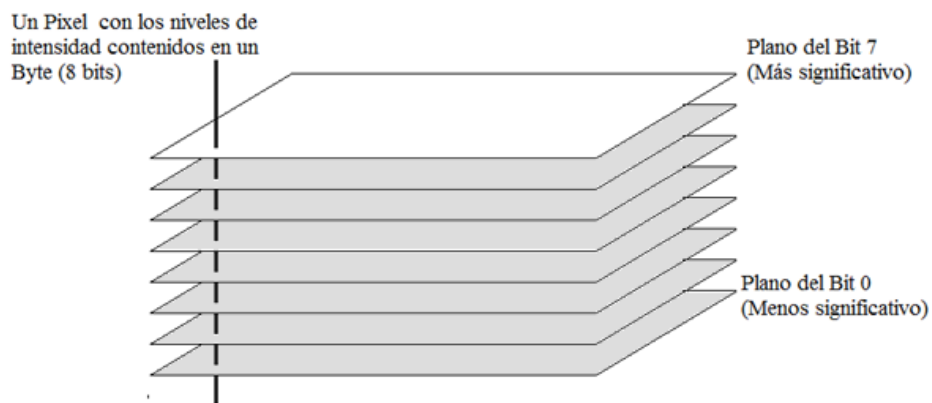


Figura 3. 9 Planos de bits

Si la imagen está dada en escala de grises de $\tau - bit/pixel$, entonces los niveles de intensidad de la imagen pueden ser representados en forma general por un polinomio de base 2 [1],

$$a_{\tau-1}2^{\tau-1} + a_{\tau-2}2^{\tau-2} + \dots + a_12^1 + a_02^0 \quad (3.6)$$

donde τ representa el número de bits que componen los niveles de intensidad en una imagen en escala de grises (para nuestro caso $\tau = 8$), los coeficientes del polinomio a_τ pueden ser valores entre cero y uno, que al multiplicarlos por potencias de base 2 indican el valor de ese coeficiente, por lo tanto, la suma de todos los términos en el polinomio es igual al valor de intensidad del pixel de interés.

Basados en esta propiedad, un método simple para descomponer la imagen en una colección de imágenes binarias es separar los τ coeficientes del polinomio en τ planos de un bit. Por lo tanto, el plano de bit de orden 0, es generado por la colección de los bits a_0 de cada pixel, mientras que el plano de bit 7 ($\tau - 1$) contiene los coeficientes $a_{\tau-1}$, por lo que cada plano de bits, contiene los coeficientes correspondientes de orden τ . La separación de los planos de bits se realiza al enmascarar el bit que genera el plano deseado mediante el uso de una operación lógica *AND* [1], [2], [3], La Tabla 3.1 muestra los valores en decimal o binario correspondientes a cada plano.

Plano deseado	Binario para operación lógica								Decimal para operación lógica
Plano 7	1	0	0	0	0	0	0	0	128
Plano 6	0	1	0	0	0	0	0	0	64
Plano 5	0	0	1	0	0	0	0	0	32
Plano 4	0	0	0	1	0	0	0	0	16
Plano 3	0	0	0	0	1	0	0	0	8
Plano 2	0	0	0	0	0	1	0	0	4
Plano 1	0	0	0	0	0	0	1	0	2
Plano 0	0	0	0	0	0	0	0	1	1

Tabla 3. 1 Valores binarios y decimales para realizar la operación lógica *AND* de los planos deseados

La Figura 3.10 muestra la contribución realizada por cada plano de bits a una imagen, sin embargo los planos 0, 1, 2, 3 y 4 presentaran valores de intensidad 1, 2, 4, 8 y 16 respectivamente (en el caso de que la operación lógica *AND* de un resultado verdadero), pero estos valores al representarse en una imagen en escala de grises contenidos en un Byte, mostraran cuadros negros, esto se debe al hecho de que para representar un nivel gris muy oscuro, los valores son cercanos al cero, en tanto que para representar el color negro utilizamos el cero, por esta razón en los primeros cinco planos de la imagen se realizan modificaciones para mostrar valores aproximados a un nivel de intensidad 100 en los casos donde la prueba lógica fue verdadera, para los otros planos no se realiza dicha modificación, pues en ellos es posible apreciar la contribución realizada por dicho plano.

Con la contribución de cada plano de bits se pueden generar nuevas imágenes adicionando o quitando planos, con la finalidad de reducir la detección de movimiento no significativo, la Figura 3.11 muestra algunas de estas imágenes. Al quitar gradualmente los planos de bits menos significativos, la imagen no parece alterarse, sin embargo, determinadas zonas parecen más homogéneas, por lo que es necesario decidir que planos de bits serán utilizados para formar la imagen que será procesada para realizar la detección de movimiento.

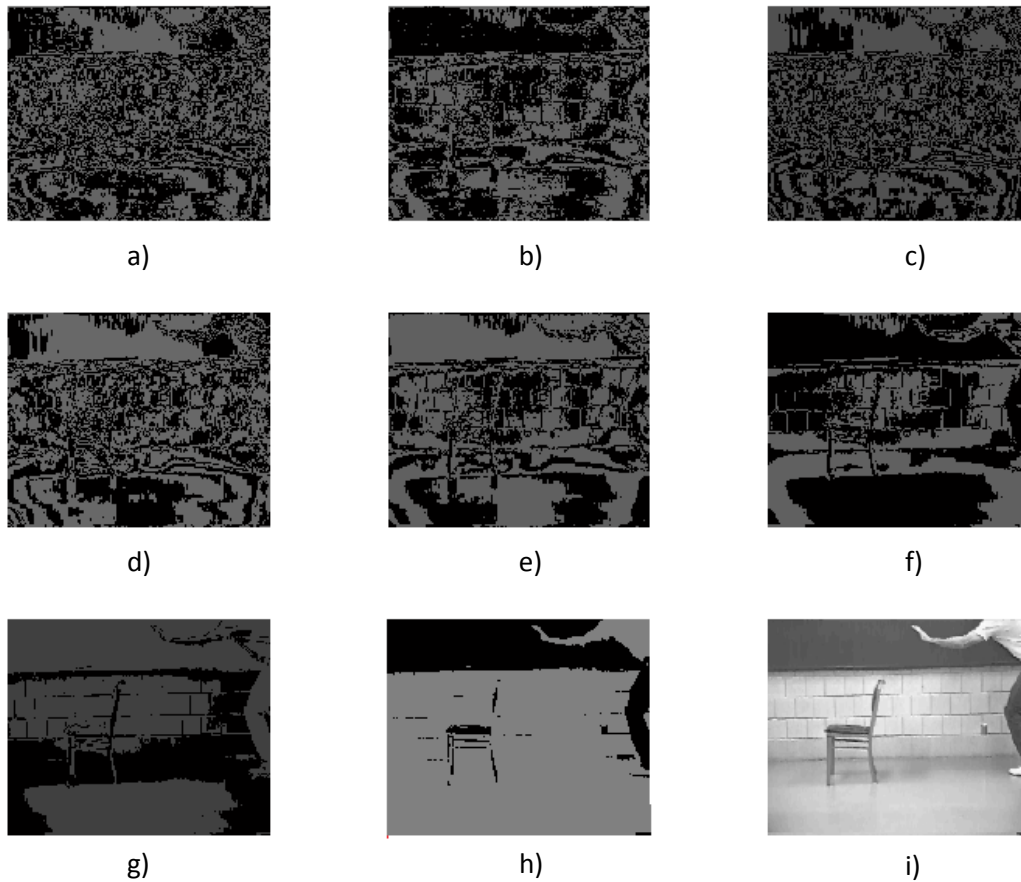


Figura 3. 10 Planos de bits de la Trama 35 del video “Chair”, a) plano 0, b) plano 1, c) plano 2, d) plano 3, e) plano 4, f) plano 5, g) plano 6, h) plano 7, i) imagen original.

Si se genera una imagen con los 2 o 3 planos de bits más significativos obtenemos imágenes con zonas homogéneas, si implementamos el **Algoritmo B** con las imágenes generadas, observamos una importante ventaja, pues la imagen de fondo permanece constante y la detección de los contornos en movimiento es más efectiva, sin embargo también se genera un problema debido a que la *MAD* representa un índice de homogeneidad de los niveles de intensidad de los píxeles que componen la ventana de análisis sobre la que se está calculando, dado que la imagen y objetos en movimiento presentan zonas homogéneas, el cálculo de las *MAD* y su posterior resta tendrá como resultado valores cercanos a cero, aún en ventanas de análisis donde exista movimiento significativo, el problema es conocido como “Problema de Apertura”. Consideremos la Figura 3.13 donde se muestra una imagen con una ventana de análisis en forma circular y un objeto rectangular que se encuentra desplazándose en línea diagonal, dado que el objeto que se está moviendo es similar en toda su estructura [11], [15], el propio sistema de visión humano tendría problemas para determinar si el objeto se está moviendo, y dado que el **Algoritmo B** se basa en

los cambios en la homogeneidad del nivel de intensidad en las ventanas de análisis, se tendrá una detección nula del movimiento, pues las ventanas de análisis entre trama y trama son similares.

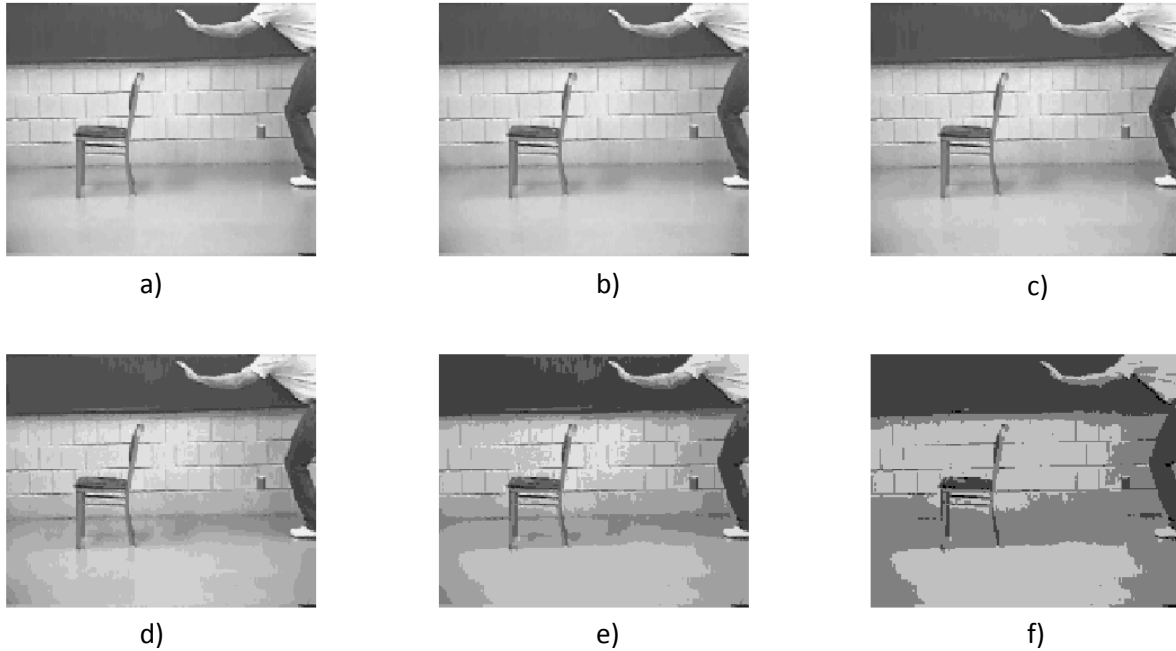


Figura 3. 11 Imágenes generadas por la suma o resta de los planos de bits. a) Imagen original menos el plano 0, b) Imagen original menos los planos 0 y 1, c) Imagen original sin los planos 0,1 y 2, d) Imagen original menos los planos 0,1,2 y 3, e) Imagen generada por la suma del plano 5, 6 y 7, f) Imagen generada por la suma de los planos 6 y 7.

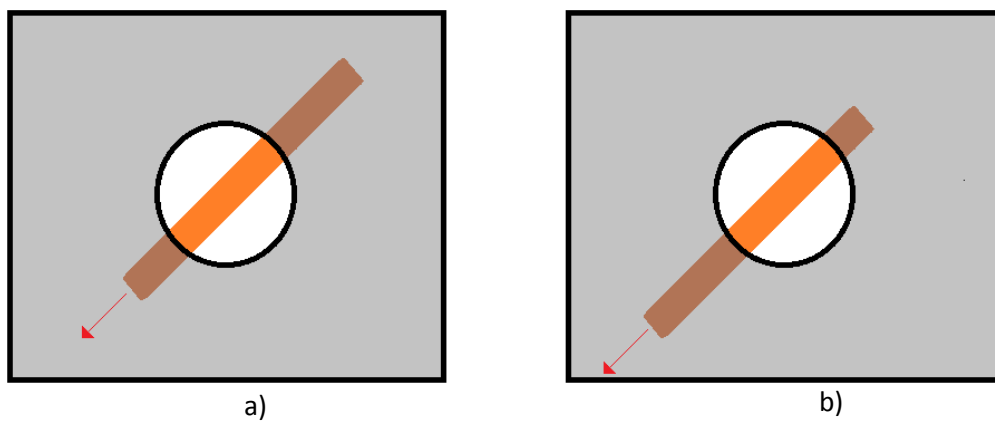


Figura 3. 12 a) desplazamiento de una barra en el tiempo t , b) desplazamiento de una barra en el tiempo $t + \Delta t$ que muestra el principio básico del “Problema de Apertura”

El ejemplo de la Figura 3.12 muestra el principio básico del problema de apertura en imágenes creadas con la computadora, en nuestro caso puede ser visto en escenas reales, por ejemplo, la Figura 3.13 muestra una imagen generada con los planos 6 y 7 de la trama 25 contenida en la secuencia de video “Chair”, donde existen regiones que son homogéneas dentro de la imagen y particularmente en el asiento de la silla se observa esa homogeneidad, así cuando se le aplica la MAD los resultados indican un valor de cero, pero el objeto se esta desplazando en línea recta horizontal justamente en el índice de columna diez, de las ventanas de análisis lo cual no solo afecta a una ventana de análisis, si no que en al menos dos de las ventanas de análisis que abarcan el asiento de la silla no serán consideradas como movimiento significativo, por esta razón, se determino generar las imágenes a procesar quitando los 4 planos de bits menos significativos, para poder reducir la detección de movimiento no significativo y a la vez reducir el efecto de apertura que ignora el movimiento significativo.

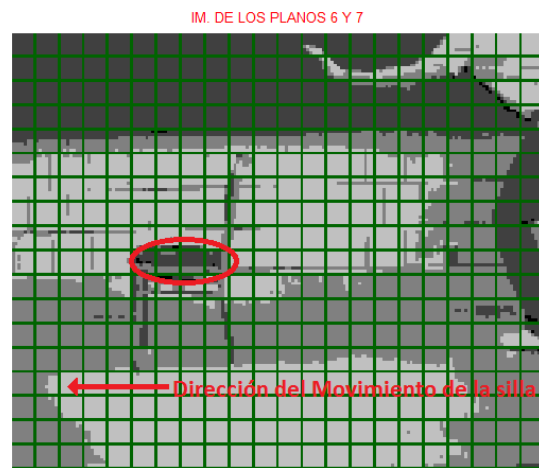


Figura 3. 13 Problema de apertura en una la Trama 25 del video “Chair”.

Ahora consideremos el problema de detección de movimiento no significativo generado por ruido natural como cambios de iluminación, movimiento de la naturaleza (en ambientes exteriores en donde existan, por ejemplo movimientos de hojas de arboles a causa del viento etc.), sombras y objetos pequeños. Estas variaciones se presentan generalmente en los bordes de objetos pequeños como el caso de las hojas, para eliminar estos movimientos se requiere realizar un emborronamiento que permita difuminar los bordes de los objetos, de este modo los objetos pequeños que presentan ligeros movimientos no causaran detecciones no significativas, para realizar esta tarea se toma como base la ecuación (3.4) que representa una función de transformación espacial donde el operador esta definido sobre una vecindad alrededor de un determinado pixel central [13].

Para realizar un emborronamiento de la imagen es necesario aplicar una operación de filtrado [26], la cual consiste en una ventana y una operación predefinidas que se deslizan sobre la imagen a esta ventana la denotamos como “la vecindad”, pues cada pixel de la imagen original será

considerado como pixel central y el resto de los pixeles de la ventana serán considerados la vecindad. El filtrado crea un nuevo pixel con coordenadas iguales a las coordenadas del pixel central de la vecindad, pero su valor es igual al resultado del filtrado [1], [3],

$$\beta(f, c) = \sum_{o=-\eta}^{\eta} \sum_{p=-\varphi}^{\varphi} \omega(o, p) \alpha(f + o, c + p) \quad (3.7)$$

,donde (f, c) son los índices de filas y columnas de la imagen original, y dado que la imagen resultante $\beta(f, c)$ tiene el mismo tamaño que la imagen original por lo tanto corresponden los índices, sin embargo para poder efectuar la operación de filtrado es necesario desplazar los índices de la ventana que contiene la operación predefinida $\omega(o, p)$, donde (o, p) son los índices necesarios para el desplazamiento para la vecindad en la imagen original, es decir $\alpha(f + o, c + p)$, para una descripción grafica de la implementación véase la Figura 3.14 y la Figura 3.15. El procedimiento de filtrado requiere de una “máscara” o ventana con los coeficientes apropiados a para alcanzar el efecto deseado, por ejemplo, suavizar la imagen, eliminar ruido, realzar bordes, etc., en nuestro caso se requiere una ventana que corresponde a un filtro pasa-bajas, es decir, que se enfatizan las bajas frecuencias y eliminan las altas frecuencias, suavizando las imágenes y suprimiendo ruidos [16], ello produce un emborronamiento de los bordes, perdiéndose nitidez visual de la imagen pero ganando en homogeneidad.

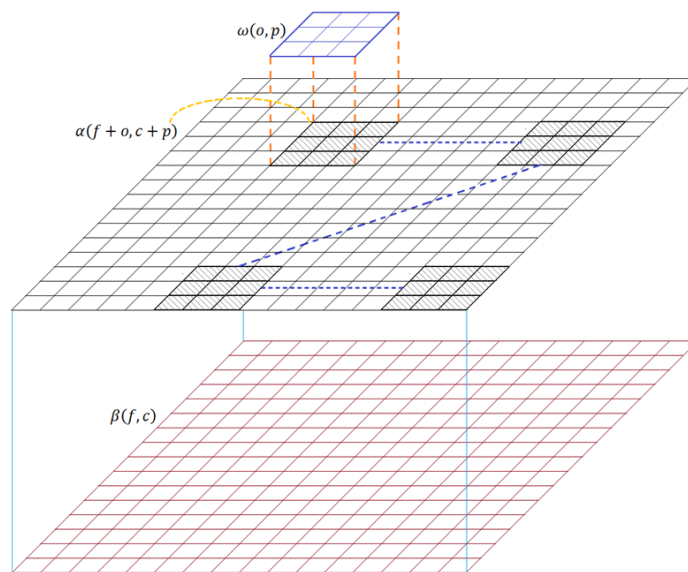


Figura 3. 14 Mecanismo de filtrado espacial.

Consideremos una ventana con u -filas y v -columnas, si se desea que la ventana tenga un solo pixel central se debe considerar que tanto el valor de u y v sean impares, además si hacemos que $u = v$ el pixel central será equidistante con los límites superiores e inferiores de la ventana, teniendo así una ventana cuadrada, por lo que en nuestro caso utilizamos $u = v = \{3,5,7\}$ y para la implementación decidimos utilizar el valor de 3, es decir, una ventana de 3x3, finalmente para complementar la ecuación (3.7) tenemos que,

$$\eta = \varphi = \frac{v - 1}{2} = 1 \quad (3.8)$$

En base a las pruebas realizadas con las ventanas encontradas en la literatura se eligieron los coeficientes de un filtro pasa-bajas Gaussiano,

$$\omega = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad (3.9)$$

La Figura 3.15 muestra un desplazamiento de los coeficientes de la ventana y los coeficientes de la vecindad alrededor del pixel de análisis (pixel central), sin embargo, al efectuar estas operaciones se presenta un problema al evaluar los pixeles en los límites, pues implica que la ventana tendrá valores que estarán fuera de la imagen que no estarán definidos y no podrá efectuarse la operación, una de las soluciones implica llenar esos pixeles faltantes con algún valor, generalmente estos valores son cero, esta solución produce efectos no deseados en los límites de las imágenes procesadas. Sin embargo, estos efectos no son significativos en tanto la ventana sea relativamente pequeña en comparación con el tamaño de la imagen procesada, como es nuestro caso, se construye una matriz mas grande que la matriz original [13] con ceros para poder efectuar la operación de filtrado [28-30].

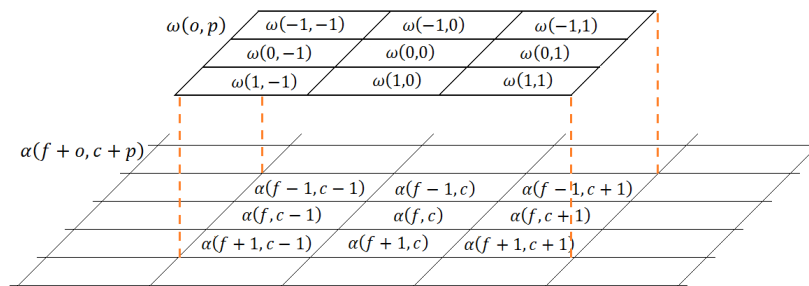


Figura 3. 15 Movimiento de los coeficientes de la ventana y los coeficientes de la vecindad con respecto al pixel central (f, c) de la imagen original.

Una vez efectuado los pre-procesamientos de enmascaramiento de los 4 bits menos significativos y el emborronamiento de la imagen se analizan los resultados para decidir que operación se realiza primero, y dado a que al emborronar la imagen difuminamos los bordes se prefirió realizar primero el emborronamiento para disminuir pequeños movimientos y posteriormente se realiza el enmascaramiento de los cuatro bits menos significativos. La Figura 3.16 muestra los resultados obtenidos por separado y las dos posibles combinaciones de los pre-procesamientos.

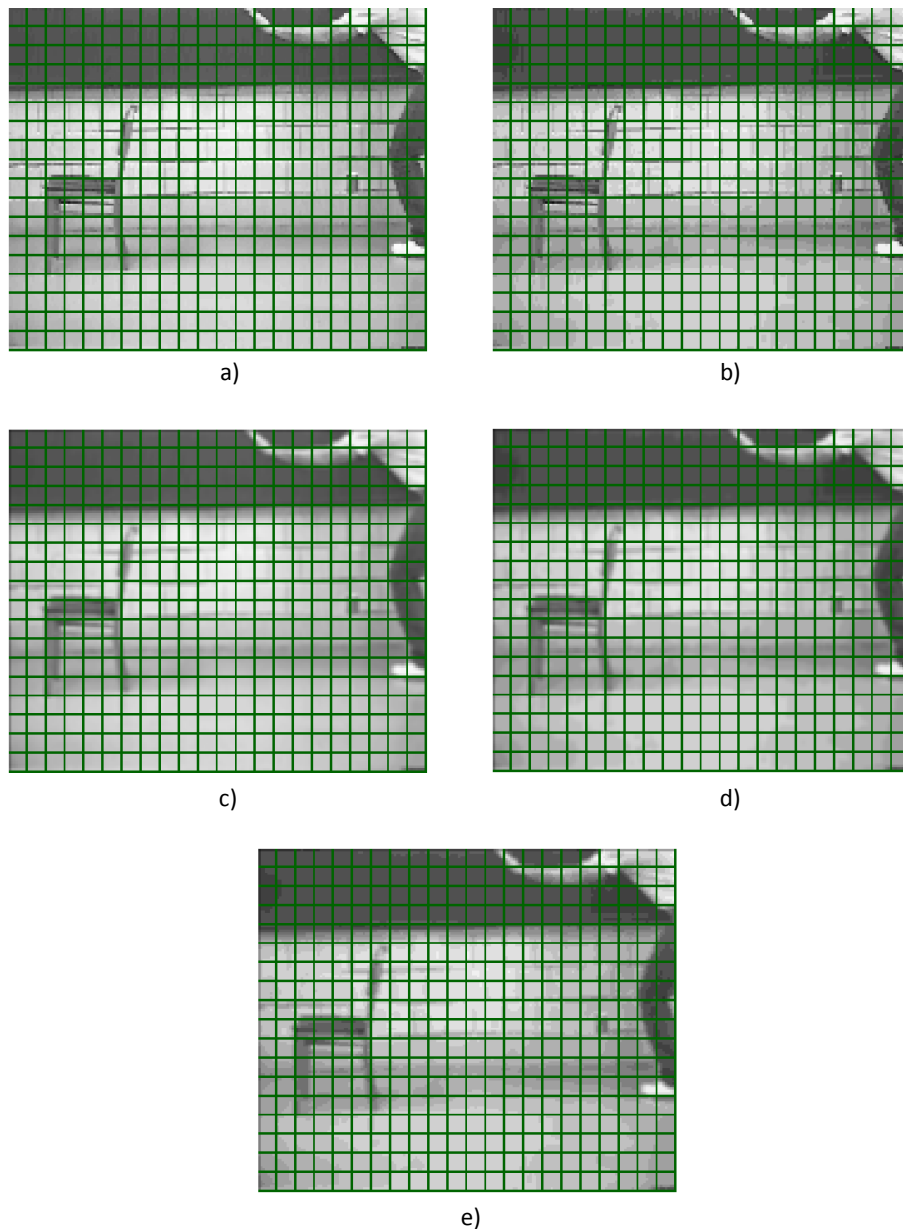


Figura 3. 16 Resultados individuales y combinados de los pre-procesamientos propuestos. a) Imagen original, b) Imagen original menos los planos 0, 1, 2 y 3, c) Imagen emborronada, d) Enmascaramiento de los 4 LSB + Emborronamiento, e) Imagen emborronada + Enmascaramiento de los 4 LSB.

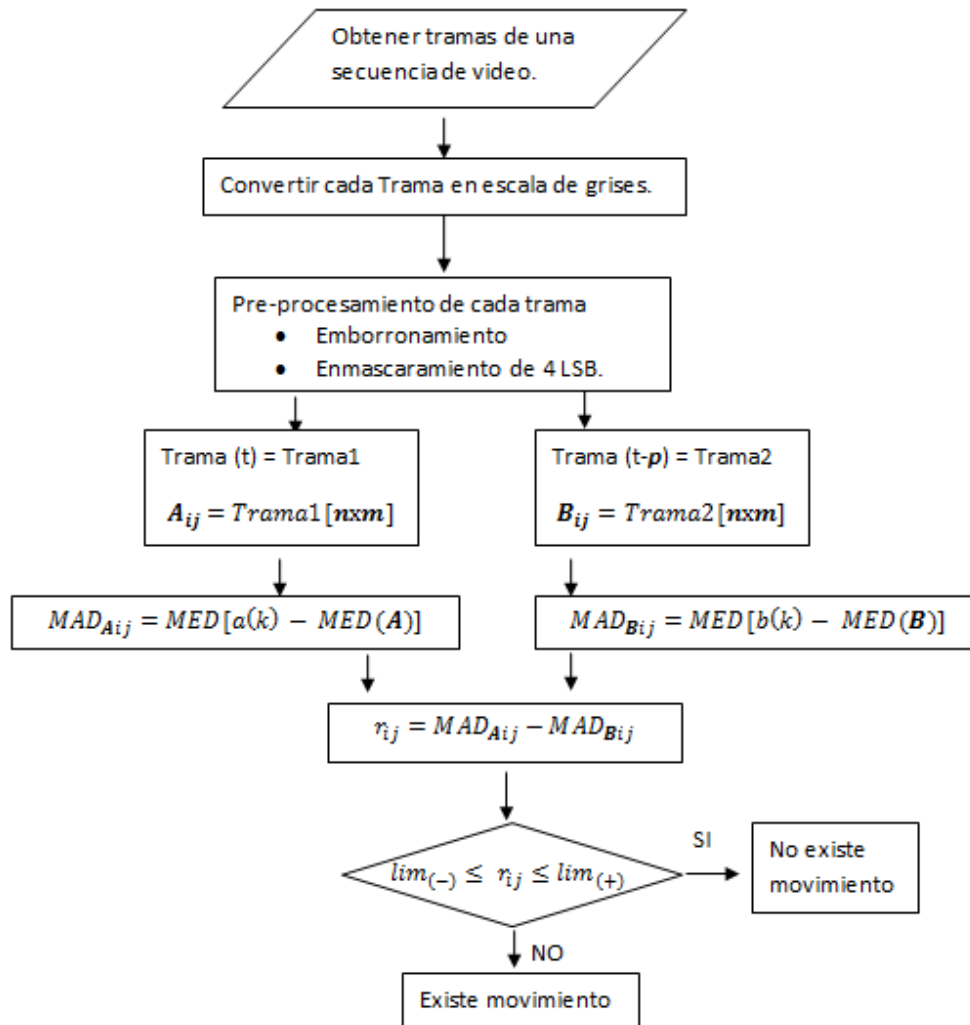


Figura 3. 17 Diagrama de flujo del *Algoritmo C*.

Finalmente, se presentan los resultados de aplicar los algoritmos propuestos a las tramas 10 y 11 de la secuencia de video "Chair", en el **Algoritmo A** se observa que existe una alta detección de movimientos no significativos, y el **Algoritmo B** muestra una notoria disminución de la detección de movimiento significativo, sin embargo, algunas ventanas de análisis donde se encuentra movimiento significativo ya no son consideradas como tales, el **Algoritmo C** mantiene una mejor detección de movimientos significativo de los objetos de interés, sin embargo aun existen algunas ventanas que no corresponden al movimiento significativo y que han sido consideradas como movimiento significativo.

Durante la realización de las pruebas del **Algoritmo C** se planteo la posibilidad de realizar el análisis de las tramas con mayor separación temporal, es decir, se considera la trama actual como la Trama1 = $trama(t)$ y la Trama2 = $trama(t - p)$ donde p se designa como el paso entre tramas, esto nos permite mejorar el desempeño del **Algoritmo C**, pues el movimiento es más evidente debido a que el desplazamiento de los objetos es mayor, aun cuando esto brinda una ventaja significativa para realizar la detección de movimiento, las modificaciones al **Algoritmo C** tendrán un paso entre tramas igual a uno, con la finalidad de que en un trabajo futuro el algoritmo final pueda ser implementado en tiempo real.

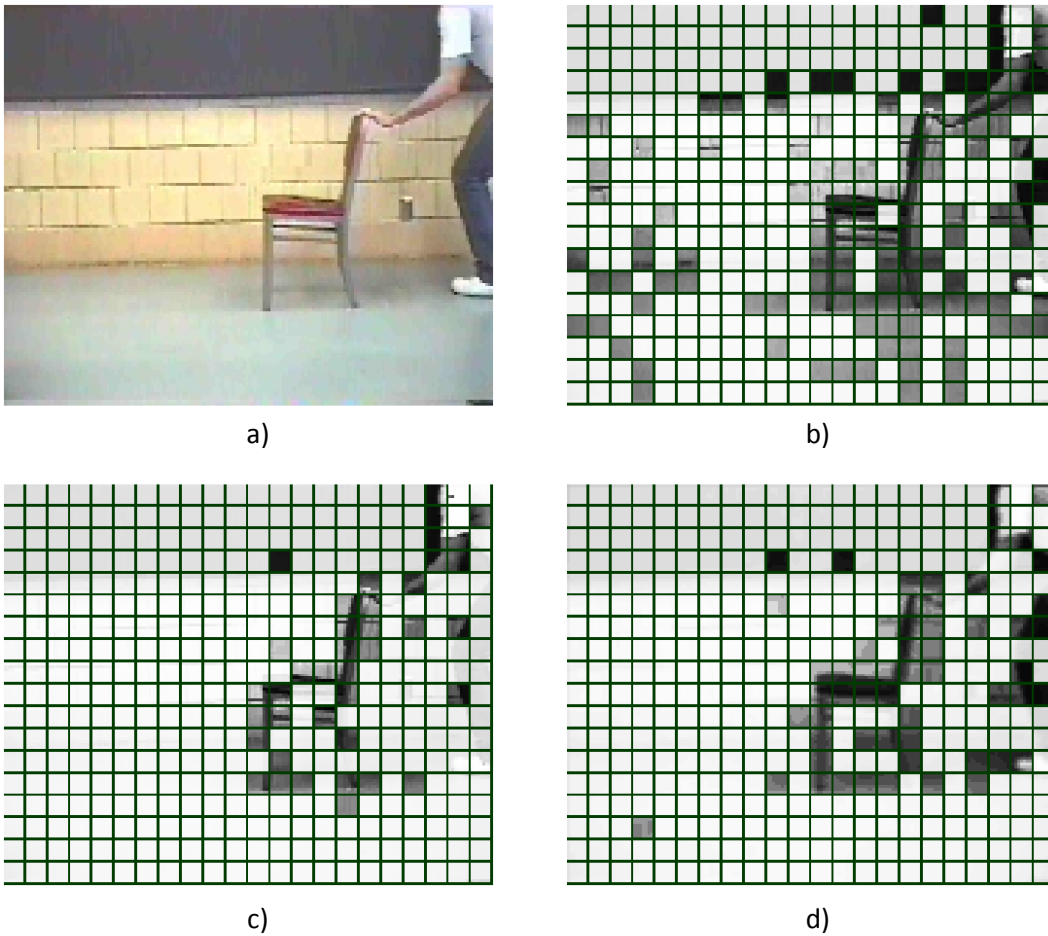


Figura 3. 18 a) Imagen Original, b) Resultado del *Algoritmo A*, c) Resultado del *Algoritmo C*, d) Resultado del *Algoritmo D*. Los algoritmos fueron aplicados a las tramas 11 y 10 de la secuencia de video "Chair".

3.5 **ALGORITMO D:** IMPLEMENTACION DEL ALGORITMO KNN Y OPERACIONES MORFOLOGICAS

En los algoritmos propuestos *A*, *B*, y *C* se consideran ventanas de análisis de 8x8 pixeles, pues se pensó realizar operaciones en el dominio de la frecuencia usando la Transformada Coseno Discreta, DCT que considera ventanas de 8x8 pixeles para desarrollar esta transformación. Ahora que se lograron buenos resultados en el dominio espacial, se decide realizar una variación del tamaño de las ventanas de análisis, además el desarrollo de los algoritmos se realiza en ambientes exteriores para poder probar el desempeño de los nuevos algoritmos. En la Figura 3.19 se presenta el procesamiento de la trama 145 usando las Tramas 144 y 145 de la secuencia de vídeo "Laura_exterior", cada trama de la secuencia de video tiene un tamaño de 240x320 pixeles, en ésta secuencia de video podemos apreciar muchos de los retos en el análisis de movimiento, por ejemplo, el movimiento de la naturaleza, iluminación, etc. Además mostramos los resultados de aplicar el **Algoritmo C**, con las ventanas de análisis de 8x8, 8x4, 4x4 y 2x4 [28], [29], [30], [31], [32]Pixeles, en estos resultados ya no se muestra la cuadrícula generada con los bordes de las ventanas de análisis, pues en ventanas de análisis pequeñas la cuadrícula no permite tener una apreciación visual de los resultados obtenidos.

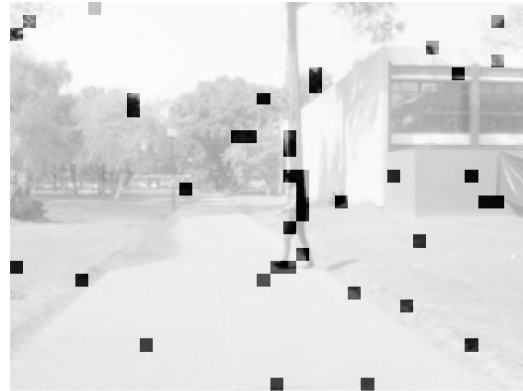
En la Figura 3.19 se observa que al disminuir el tamaño de las ventanas de análisis las detecciones del movimiento son más concentradas alrededor del objeto de interés, sin embargo existen movimientos no significativos que no se han logrado disminuir. Para eliminar esta problemática se propone implementar el algoritmo KNN (K-Nearest Neighbors), [23] el cual es usado generalmente para extraer información de una vecindad alrededor de un determinado pixel central.

Hasta el momento las imágenes de la Figura 3.19 son una adaptación de los procesamientos realizados para una apreciación visual de la implementación de los algoritmos, pues presentamos imágenes con modificaciones de contraste para notar las ventanas en las que se realizó una detección, sin embargo las imágenes resultantes son imágenes binarias, es decir en ventanas donde existe movimiento tenemos un valor de uno, mientras que las ventanas que no representan movimiento tienen un valor de cero, estos valores están almacenados en el arreglo r_{ij} esto nos permite realizar una adaptación al algoritmo KNN [23].

Originalmente el algoritmo KNN [23], [27] involucra el cálculo de las distancias Euclidianas desde el pixel central hasta cualquiera de los pixeles de la vecindad para determinar el nivel de relación que ambos pixeles tienen, sin embargo esto involucra un elevado costo computacional y no es determinístico para el cálculo que se desea realizar, para la implementación del algoritmo KNN propuesto nos basaremos en los conceptos de filtrado tratados en la sección 3.3.



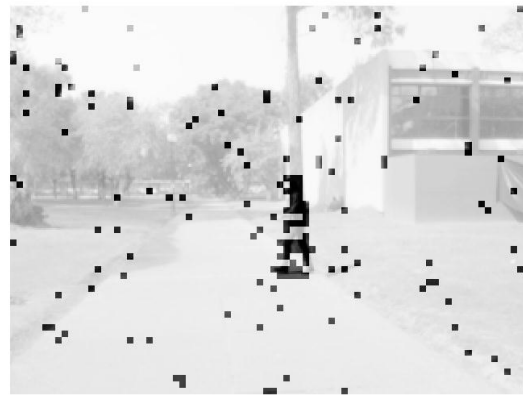
a)



b)



c)



d)



e)

Figura 3. 19 a) trama 145 en espacio RGB. Resultados del *Algoritmo C* aplicado a las tramas 144 y 145 de la secuencia de video "Laura_exterior" utilizando: b) Ventanas de análisis de 8x8 píxeles, c) Ventanas de análisis de 8x4 píxeles, d) Ventanas de análisis de 4x4 píxeles, e) Ventanas de análisis de 2x4 píxeles.

La primera consideración es el tamaño de la ventana que determina la “vecindad” para la implementación del KNN, el número de filas y columnas deben ser iguales e impares y los valores propuestos son: 3x3, 5x5 y 7x7, en la Figura 3.19 e), muestra mejores resultados al usar ventanas de análisis de 2x4 pixeles, esto implica que el arreglo r_{ij} tiene mas elementos en comparación con otros tamaños de ventanas de análisis, con esta consideración se decidió utilizar una vecindad de 7x7 pixeles para analizar el comportamiento del del pixel central con su vecindad. Una vez seleccionado el tamaño de las ventanas nos enfrentamos nuevamente a la problemática de evaluar el comportamiento de los pixeles en los bordes de la imagen, y al igual que en el **Algoritmo C** se decidió utilizar una matriz aumentada en los bordes y rellena con valores de ceros la Figura 3.20 ilustra este proceso, esto no afecta el análisis del comportamiento de la vecindad con respecto al pixel central [27], [28], [29], [30].

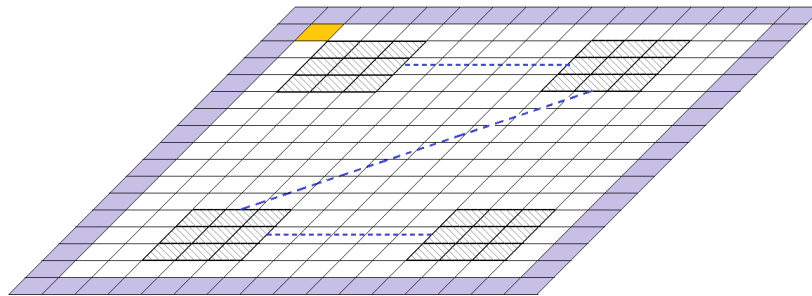


Figura 3. 20 Desplazamiento de las ventanas para realizar la implementación del Algoritmo KNN propuesto

Analizar el comportamiento de los vecinos cercanos con respecto al pixel central nos permite eliminar los valores considerados movimientos significativos [23], [25], y como mostramos en la Figura 3.19 el movimiento que es significativo se encuentra concentrado en el objeto de interés, con este argumento proponemos un algoritmo que genere una matriz aumentada para procesar las vecindades de 7x7 pixeles, posteriormente cada elemento del nuevo arreglo r_{ij} aumentado, será considerado como pixel central para el análisis de la vecindad [27], si el valor del pixel central es igual a uno, entonces se realizara una suma de los pixeles de la vecindad, si la suma es mayor o igual a $umbral_knn$, proponemos que ese pixel central puede ser considerado como movimiento, y en caso contrario asignarle el valor de cero, es decir que no represento un movimiento significativo, el $umbral_knn$ que se selecciono fue determinado en base a diversas pruebas y principalmente a que permitiera preservar la mayor parte del movimiento significativo y eliminara la mayor parte del movimiento no significativo, el umbral puede seleccionarse en un intervalo desde uno hasta cuarenta y nueve que es el numero de vecinos considerados por la ventana, si el valor es muy grande, por ejemplo veinte, se eliminaran todos los movimientos que no son significativos pero también eliminaría elementos en los que realmente existe movimiento, de manera opuesta, si el valor del $umbral_knn$ es muy pequeño, por ejemplo tres, no eliminaría todos los movimientos no significativos, aunque preservaría mejor los elementos de r_{ij} que

representan movimiento significativo, después de diversas pruebas, se considero $umbral_{knn} = 7$.

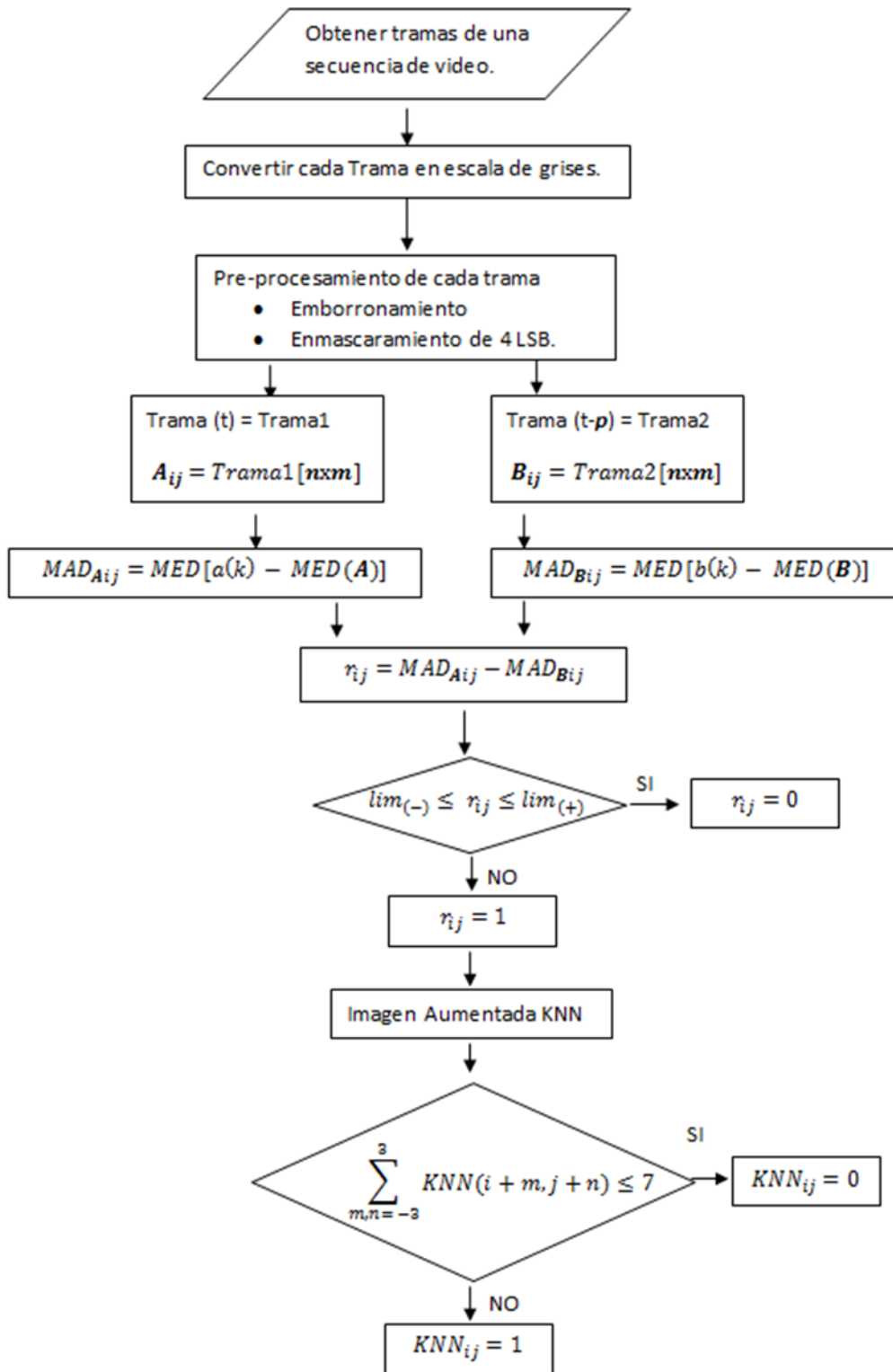
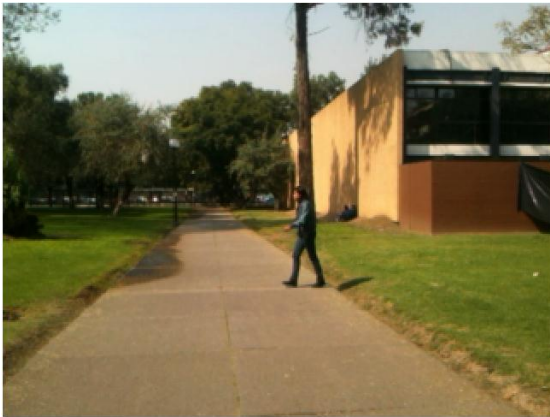


Figura 3. 21 Diagrama de flujo del Algoritmo D, Implementación de un algoritmo KNN



a)



b)



c)



d)

Figura 3. 22 a) trama 145 de la secuencia de video “Laura_exterior”. Aplicación del algoritmo KNN con: b) $umbral_knn = 3$, c) $umbral_knn = 20$, d) $umbral_knn = 7$.

En algunas imágenes procesadas se observa que el objeto de interés tiene pequeñas regiones separadas del objeto principal pero que forman parte del objeto, por ejemplo en el caso de personas se observa que las extremidades pueden presentar problema. Para poder seguir un objeto o identificar si existe más de un objeto en la escena, es necesario que estos tengan unidas todas las regiones que lo componen [19].

Para poder unir las pequeñas regiones del objeto que se encuentran separadas se hace uso de las operaciones morfológicas matemáticas, que forman parte de una teoría para el análisis de la forma de objetos basados en las operaciones y teorías de conjuntos [21].

Consideremos que X es un plano Euclidiano bidimensional y que A y B son subconjuntos de X , con esta notación podemos describir una imagen binaria, es decir una imagen cuyos valores son cero y uno, generalmente en imágenes binarias se considera que en el valor de cero existe la información de la imagen y por ello se toma para el procesamiento se considera la ubicación en el plano Euclidiano [20] de dichos valores que también llamaremos pixeles, en nuestro caso la información de la imagen se encuentra representada en los valores de uno, y el fondo será considerado cero.

La suma de Minkowski de $A \subseteq X$ y $B \subseteq X$, denotada por $A \oplus B$ es el conjunto resultante de sumar cada elemento de A con cada elemento de B , es decir,

$$A \oplus B = \{a + b | a \in A \wedge b \in B\} \quad (3.10)$$

A esto se le conoce como la operación de dilatación de un conjunto A (en nuestro caso es una imagen binaria), por un conjunto B que representa al elemento estructural, y esencialmente sirve para adicionar pixeles, ampliar bordes y unir objetos próximos.

La resta de Minkowski de $A \subseteq X$ y $B \subseteq X$, denotada por $A \ominus B$ cuya definición es:

$$A \ominus B = \{x \in X | x + b \in A, \forall b \in B\} \quad (3.11)$$

Esta operación se conoce como la operación de erosión de A por B , su utilidad consiste en eliminar pixeles, reducir bordes y separar objetos próximos, a pesar de que la dilatación y la erosión son operaciones básicas independientes, dentro de la morfología matemática se usan por parejas alternadas; es decir, normalmente se realizan procesos iterativos del tipo erosión seguida de una dilatación y una dilatación seguida de una erosión, estas operaciones se denominan apertura y cerradura, respectivamente [19], [20], [21], [22]. Dado que la erosión y la dilatación no son operaciones inversas, los resultados son distintos, y de hecho se consideran como operaciones fundamentales (básicas) de la morfología matemática, sin embargo su definición esta dada en términos de la dilatación y la erosión.

En nuestro caso usamos una operación morfológica de cerradura $A \bullet B$, la cual consiste en la realización de una dilatación seguida de una erosión,

$$A \bullet B = (A \oplus B) \ominus B \quad (3.12)$$

Esta operación utiliza el mismo elemento estructural para la dilatación y erosión, delimitando el tamaño y la forma final de los objetos procesados, los elementos que comúnmente se utilizan son: Diamante, Círculo, Cuadrado, Rectángulo, Líneas, Octágono y en algunos casos especiales puede utilizarse una forma arbitraria.

En base a las pruebas efectuadas, obtuvimos mejores resultados visuales utilizando un elemento estructural de una línea con longitud de 15 pixeles, de un pixel de ancho y un ángulo de 45 grados de inclinación, los elementos estructurales están contenidos en un arreglo matricial [25], la figura 3.23 muestra el arreglo del elemento estructural usado, esto nos permite unir los elementos que representan a un solo objeto en movimiento y que debido al procesamiento previamente realizado el objeto pudo haber sido separado.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura 3. 23 Elemento estructural usado para las operaciones morfologías, una línea de 15 pixeles de longitud, 1 pixel de ancho y 45 grados de inclinación.

La Figura 3.24 muestra los resultados de aplicar el **Algoritmo D** y los resultados obtenidos al realizar la operación de cerradura a los resultados del **Algoritmo D**.

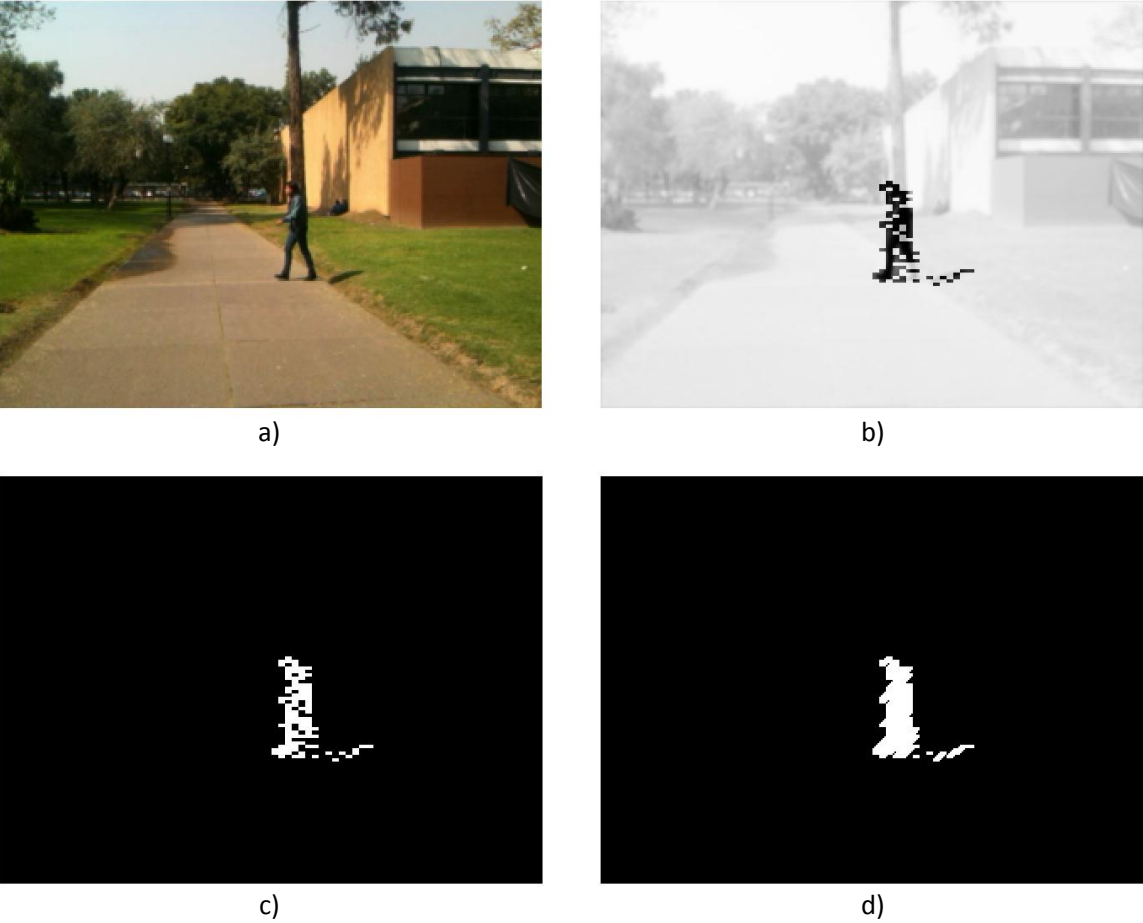


Figura 3. 24 a) trama 145 en el espacio RGB, b) Modificación del contraste para señalar el movimiento detectado, c) Imagen binaria del movimiento, d) Imagen binaria del movimiento después de aplicar la operación morfológica de cerradura.

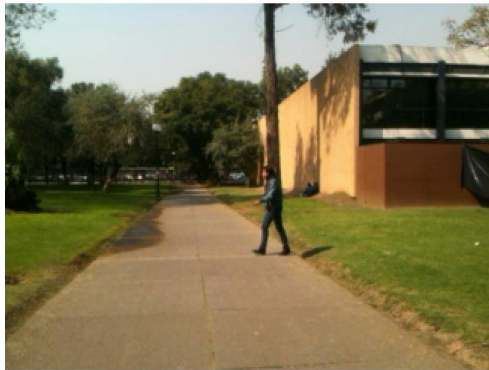
Ahora tenemos la posibilidad de detectar uno o más objetos en la escena, y es importante determinar la prioridad del objeto que debe ser considerado, si el objeto se acerca a la cámara o se mueve más rápido que los demás objetos en la escena, son parámetros que permitirán considerarlo como prioritario [14].

Para implementar estas necesidades consideremos que el objeto en movimiento tiene un área proporcionada por el conjunto de píxeles que representan el movimiento, si un objeto se está más cercano a la cámara que los otros objetos en movimiento, este presentará una mayor área de movimiento. Por otro lado, si el objeto se encuentra desplazándose más rápido en la escena que los otros objetos, también generará mayor área de movimiento, por lo que para asegurar la prioridad del movimiento nos basaremos en el área generada por el objeto, considerando como prioritaria el objeto de mayor área, si existen más objetos en la escena todos pueden ser considerados al mismo tiempo como objetos en movimiento, siempre que los otros objetos tengan una área de movimiento similar a la del objeto de mayor área de movimiento [24], para cumplir estas condiciones se propone el uso de $umbral_{\text{área}} = \text{área}_{\text{máxima}} \times 0.6$ el valor 0.6 se eligió dado que en una distribución normal de los datos representa el valor característico del punto de inflexión, y con las pruebas realizadas se consolidó como una opción que permitiría marcar dos o más objetos en movimiento en la escena, pero que también permite darle prioridad a los objetos con mayor área de movimiento.

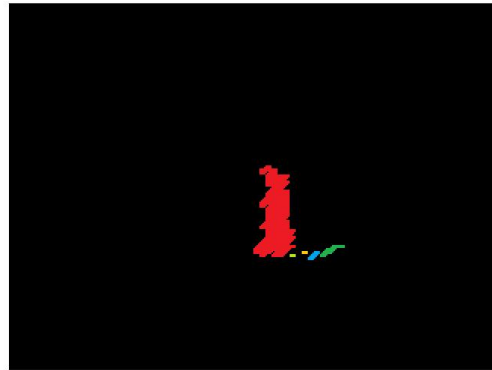
Sin embargo si los objetos tienen menor área que la calculada con el $umbral_{\text{área}}$, entonces estos objetos son eliminados, lo que nos permite discriminar objetos pequeños y que no presentan movimiento significativo, si consideramos los resultados mostrados en la Figura 3.24 c), observamos que el objeto en movimiento proyecta una sombra, la cual durante la aplicación del algoritmo fue separada del objeto, y como esta no presenta movimiento significativo puede ser eliminada.

En la Figura 3.25 b) se muestran cinco objetos de movimiento que han sido coloreados para mostrar que el sistema está considerando solamente al objeto de mayor área de movimiento.

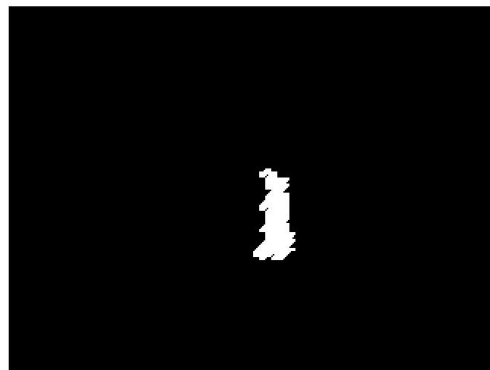
Finalmente el objeto que presenta un movimiento significativo se enmarca con el rectángulo lo más pequeño posible que contenga todos los píxeles (Bounding Box) de tal objeto, además se utiliza un puntero para determinar el centroide de dicho rectángulo con la finalidad de proporcionar una mejor apreciación de los objetos detectados [32]. Para tener una mejor apreciación visual de los resultados [24], estas operaciones se muestran sobre la imagen en el espacio RGB, y esta representación será considerada como el resultado final del **Algoritmo D**.



a)



b)



c)

Figura 3. 25 a) Trama 145 en espacio RGB, b) Diversos objetos en movimiento, c) Aplicación $umbral_{\acute{a}rea}$.

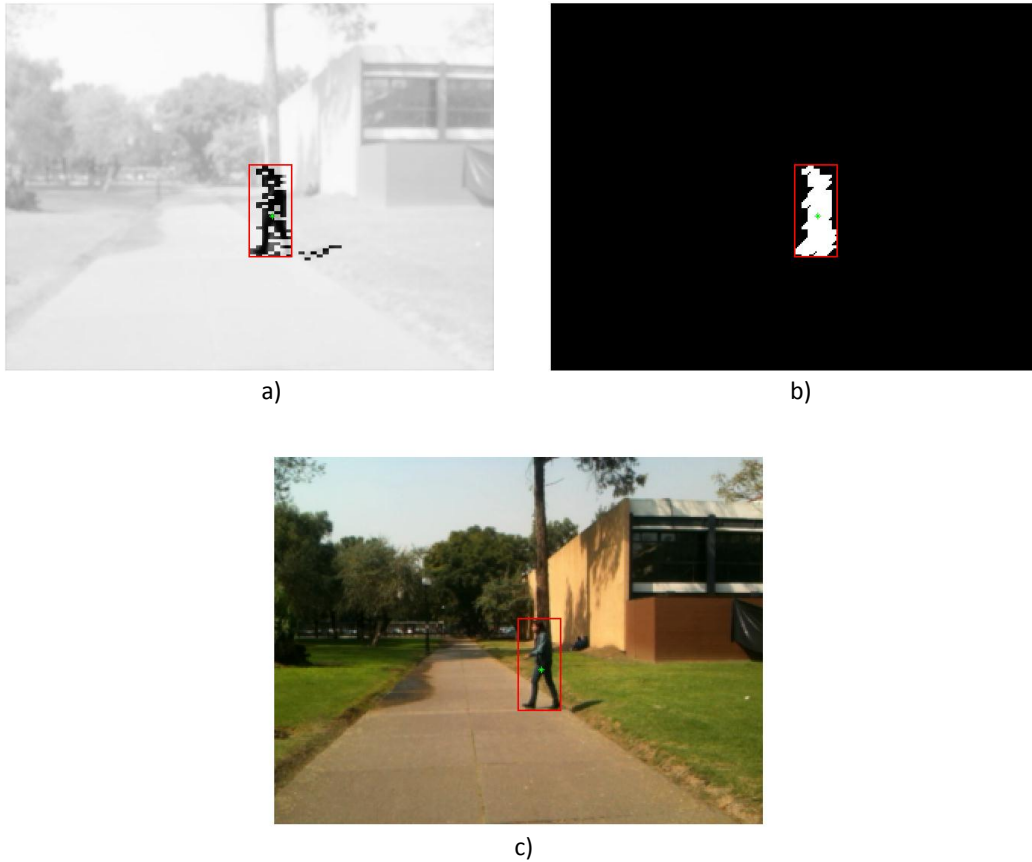


Figura 3. 26 Enmarcado y centroide del objeto en movimiento, a) En la imagen con modificación de contraste, b) En la imagen binaria después de la operación de cerradura, c) En la imagen en espacio RGB.

3.6 CONCLUSIONES

En este capítulo se mostro el desarrollo de los algoritmos propuestos desde el uso de la MAD en secuencias de video simuladas, la propuesta e implementación de un umbral dinámico en la resta de las MAD para discriminar movimiento significativo, la propuesta e implementación de pre-procesamientos para emborronar las imágenes y el enmascaramiento de los 4 LSB para disminuir la detección movimiento no significativo, las variación del tamaño de las ventanas de análisis, la implementación de un algoritmo KNN y la propuesta del `umbral_knn` que determina si la detección de movimiento corresponde aun objeto de interés y la aplicación de operaciones morfológicas que agrupan las regiones que pudieran separarse durante el proceso de detección de movimiento, además la propuesta de un `umbral_área` para el seguimiento del objeto con mayor área de movimiento y la detección de dos o más objetos con áreas similares, finalmente mostrar sobre la imagen en espacio RGB un rectángulo y centroide sobre los objetos en movimiento que se han detectado.

3.7 REFERENCIAS BIBLIOGRÁFICAS

- [1] R.C. Gonzalez, R.E. Woods, *Digital Image Processing, 3rd Ed.*, New Jersey, USA: Prentice Hall, Inc., 2008, ISBN-13: 9780131687288.
- [2] R. C. Gonzalez, R. E. Woods and S. T. Eddins, *Digital Image Processing using Matlab. 2nd Ed.*, Knoxville, USA: Gatesmark Publishing, 2009, ISBN-13: 9780982085400.
- [3] W. K. Pratt, *Digital Image Processing, 4th Ed.*, New Jersey, USA: Jhon Wiley & Sons, Inc., 2007, ISBN-139780471767770.
- [4] Hoaglin, D.C., Mosteller, F., and Tukey, J.W., *Understanding Robust and Exploratory Data Analysis*. John Wiley & Sons, 1983, ISBN-13: 978-0471384915.
- [5] G. P. Martinsanz, J. M. de la Cruz Garcia, *Visión por computador: Imágenes digitales y aplicaciones*, Alfaomega Grupo Editor (2002), ISBN: 9701508041.
- [6] W. W. Hines, *Probability and Statistics in Engineering*, John Wiley (2003), ISBN: 9780471240877.
- [7] R. H. Frank, *Robust statistics: the approach based on influence functions*, Wiley (1986), ISBN: 9780471829218.
- [8] Heikkilä, M., and Pietikäinen, M., "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(4), 657-662. 2006.
- [9] Su, J., Liu, Q., and Ikenaga T., "Motion detection based motion estimation algorithm for video surveillance application," *IEEE Int. Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2009)*, 481-484, 2009.
- [10] Cheng, F.C., Huang, S.C., and Ruan, S.J., "Scene analysis for object detection in advanced surveillance systems using laplacian distribution model," *IEEE Trans. Systems, Man, and Cybernetics –Part C: Applications and Reviews*, 41(5), 2011.
- [11] Augustin, B., Sujitha, J., and Palanikumar, S., "Motion and feature based person tracking in surveillance videos," *IEEE Proc. ICETECT 2011*, 605-609, 2011.
- [12] Gil-Jimenez, P., Maldonado-Bascon, S., Gil-Pita, R., and Gomez-Moreno, H., "Background Pixel Classification for Motion Detection in Video Image Sequences", *Proc. Artificial and natural neural networks 7th international conference on Computational methods in neural modeling - Volume 1, IWANN'03*, 2003.
- [13] Aboshosha, M. Hassan, M. Ashour and M. El Mashade, "Image denoising based on spatial filters, an analytical study," *in Computer Engineering & Systems, 2009. ICCES 2009. International Conference on*, pp. 245-250 (2009).
- [14] Stauffer, C., and Grimson, W.E.L., "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(8), 747-757. 2000.
- [15] Song, Y., Feng X., Perona, P., "Towards detection of human motion," *Proc. IEEE Computer Vision and Pattern Recognition*, 810-817, Vol.1, 2000.
- [16] T. Pham-Gia and T. L. Hung, "The mean and median absolute deviations," *Mathematical and Computer Modelling* 34(7-8), 921-936 (2001).
- [17] Gallegos-Funes, F.J., and Ponomaryov, V., "Real-time image filtering scheme based on robust estimators in presence of noise impulsive," *Real Time Imaging*, 8(2), 78-90, 2004.

- [18] Gallegos-Funes, F.J., Rosales-Silva, A.J., and Toledo-Lopez, A., "Multichannel Image Processing by using the Rank M-Type L-Filter," *Journal of Visual Communication and Image Representation*, 23, 323-330, 2012.
- [19] Chao, X., Yanjun, L., and Ke, Z., "Visual motion detecting and deblurring based on mathematical morphology and ensemble learning," 6th IEEE Int. Conf. Wireless Communications Networking and Mobile Computing (WiCOM), 1-4, 2010.
- [20] R. M. Haralick, S. R. Sternberg and X. Zhuang, "Image Analysis Using Mathematical Morphology," *Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-9(4)*, 532-550 (1987).
- [21] H. Joo and R. M. Haralick, "Understanding the application of mathematical morphology to machine vision," in *Circuits and Systems, 1989., IEEE International Symposium on*, pp. 977-982 vol.972 (1989).
- [22] Meijster and M. H. F. Wilkinson, "A comparison of algorithms for connected set openings and closings," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 24(4), 484-494 (2002).
- [23] H. Samet, "K-Nearest Neighbor Finding Using MaxNearestDist," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 30(2), 243-252 (2008).
- [24] Lee, S.H., and Kang, M.G., "Motion tracking based on area and level set weighted centroid shifting," *IET Computer Vision*, 4(2), 73-84, 2010.
- [25] "R2012a Documentation, Image Processing Toolbox, Morphology Fundamentals: Dilation and Erosion" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/images/f18-12508.html>.
- [26] "R2012a Documentation, Image Processing Toolbox, Understanding Deblurring" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/images/f9-5358.html>
- [27] "R2012a Documentation, Image Processing Toolbox, Neighborhood or Block Processing: An Overview" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/images/f7-15480.html>
- [28] "R2012a Documentation, Image Processing Toolbox, Performing Sliding Neighborhood Operations" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/images/f7-12915.html>
- [29] "R2012a Documentation, Image Processing Toolbox, Performing Distinct Block Operations" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/images/f7-12726.html>
- [30] "R2012a Documentation, Image Processing Toolbox, Morphology Fundamentals, Using Columnwise Processing to Speed Up Sliding Neighborhood or Distinct Block Operations" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/images/f7-12394.html>
- [31] "R2012a Documentation, Image Processing Toolbox, Morphology Fundamentals, Working with Image Data in the MATLAB Workspace" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/imaq/f12-139775.html>
- [32] "R2012a Documentation, Image Processing Toolbox, Pixel Values and Statistics, regionprops" 1984-2012- The MathWorks, Inc. Disponible en: <http://www.mathworks.com/help/toolbox/images/ref/regionprops.html>

4.1 INTRODUCCIÓN

En esta sección se muestran los resultados obtenidos al aplicar el **Algoritmo D** a diferentes secuencias de vídeo, las cuales se tomaron en ambientes de iluminación representativos de las escenas en las que el algoritmo pudiera ser usado, el ambiente interior fue tomado en el pasillo de la SEPI-Electrónica, mientras que las secuencias de video en ambiente exterior fueron tomadas en los pasillos del Edificio Z y los Edificios 4, 5 y 6 de la UPALM.

Las secuencias de video se tomaron con diferentes dispositivos. Los resultados experimentales se agrupan en: ambientes interiores, ambientes exteriores, ambientes con objetos que se desplazan a mayor velocidad que el promedio de la velocidad de desplazamiento en humanos y ambientes no controlados.

4.2 Desarrollo

Los resultados experimentales para cada secuencia de video se presentan en 24 figuras agrupadas como una sola. Las tramas usadas fueron elegidas de manera que muestren la trayectoria de los objetos de interés.

Las figuras se agrupan en columnas para indicar las tramas que se están procesando, en tanto que los renglones considerados de arriba a abajo representan lo siguiente:

El primer renglón muestra la trama actual, es decir la trama (t) en espacio RGB. Se usa la trama actual, pero para aplicar el **Algoritmo D** es necesario también usar la trama anterior es decir la trama ($t - 1$). Por esto en las imágenes solo se presenta el número de la trama actual.

El segundo renglón muestra una imagen en escala de grises con la modificación del contraste en las ventanas de análisis donde se considera la existencia de un objeto en movimiento. La modificación del contraste permite ubicar en la escena las ventanas consideradas como movimiento y así saber si el movimiento es ocasionado por un objeto de interés, por ejemplo una persona o vehículo.

El Tamaño de las ventanas de análisis para la aplicación del **Algoritmo D** en las secuencias de video usadas en este capítulo, es de 2×4 pixeles.

El tercer renglón presenta una imagen binaria, con las regiones que son consideradas como movimiento significativo (color blanco) y en cero (color negro) el resto de la imagen.

El cuarto renglón muestra la operación morfológica de cerradura aplicada en la imagen binaria, en esta representación observamos como se conectan las ventanas de análisis de los objetos considerados movimiento significativo.

El quinto renglón muestra el objeto de mayor área relativa de movimiento, enmarcado en un rectángulo e indicando el centroide, en esta imagen se aprecia la aplicación del umbral de área de movimiento relativa, por esto, las ventanas que erróneamente fueron consideradas movimiento significativas son eliminadas. Esta imagen también permite observar como el algoritmo le da prioridad al objeto que genera mayor área relativa de movimiento y mediante el umbral de área relativa determinar si existe más de un objeto en movimiento.

El sexto renglón muestra la imagen en espacio RGB señalando mediante un rectángulo y su centroide los objetos detectados, estas imágenes son el resultado final del **Algoritmo D**.

4.3 RESULTADOS EXPERIMENTALES EN AMBIENTES INTERIORES

Las Figuras 4.1 a), b), c) y d) son la representación de un experimento donde una persona camina a una velocidad promedio de 0.83 m/s, desplazándose hacia la cámara finalizando con un desplazamiento paralelo al plano de la cámara. Esta secuencia de video fue llamada "Laura_pasillo".

En las Figuras 4.1 a) se muestran los resultados de aplicar el **Algoritmo D** a la trama 180 (trama actual). En la Figura 4.1 b) Los resultados obtenidos al aplicarlo a la trama 205. La Figura 4.1 c) los resultados obtenidos al aplicarlo a la trama 230 y la Figura 4.1 d) los resultados obtenidos al aplicarlo a la trama 250.

En el segundo renglón de las figuras, observamos que el objeto en movimiento es detectado, pero también observamos en el segundo renglón de las figuras que componen la Figura 4.1, que se detecta movimientos en zonas en las que realmente no existe movimiento, principalmente en la pared formada con vidrios. Esto se debe al reflejo de los objetos en movimiento y al de las lámparas que iluminan el pasillo.

En el cuarto renglón observamos la unión de las ventanas de análisis que se consideraron movimiento significativo y como las ventanas son agrupadas en objetos. En el quinto renglón se observa que se eliminaron los objetos que son más pequeños que $umbral_{\text{área}}$ para finalmente obtener un rectángulo que delimita el objeto de mayor área relativa de movimiento y su centroide en una imagen en espacio RGB.

Las Figuras 4.2 a), b), c) y d) son la representación de un experimento donde una persona camina a una velocidad promedio de 0.83 m/s, desplazándose hacia la cámara. Esta secuencia de video fue llamada "Mariana_pasillo". En la Figura 4.2 a) se utiliza la trama 115, en la Figura 4.2 b) se utiliza la trama 135, en la Figura 4.2 c) la trama 157 y la Figura 4.2 d) la trama 191.

Al igual que el experimento presentado en la Figura 4.1, el experimento mostrado en la Figura 4.2 detecta movimiento no significativo causado por los reflejos en los vidrios que forman una de las paredes del pasillo, sin embargo el **Algoritmo D** permite detectar solo el movimiento del objeto de interés.

4.4 RESULTADOS EXPERIMENTALES EN AMBIENTES EXTERIORES

Las Figuras 4.3 a), b), c) y d) son la representación de un experimento donde se solicitó a una señorita se desplazara de derecha a izquierda y paralelamente al plano de la cámara a una velocidad promedio de 0.83 m/s. Esta secuencia de video fue nombrada “Laura_exterior”.

La secuencia de video “Laura_exterior”, fue tomada en el pasillo exterior del edificio 5 de la UPALM, en estas imágenes observamos un ambiente exterior con ligeros movimientos de las hojas, la mayor parte de la escena permanece estática.

El objetivo de este experimento es determinar la robustez del **Algoritmo D** para detectar el movimiento de objetos en escenarios exteriores considerando un solo objeto en movimiento.

Las tramas utilizadas para la aplicación del **Algoritmo D** son: en la Figura 4.3 a) la trama 80, en la Figura 4.3 b) la trama 133, para la Figura 4.3 c) la trama 160 y para la Figura 4.3 d) la trama 255. La selección de estas tramas representa el desplazamiento del objeto en la secuencia de video.

Las Figuras 4.4 a), b), c) y d) son la representación de un experimento donde un profesor se desplaza de izquierda a derecha y transversalmente al plano de la cámara a una velocidad promedio de 0.83 m/s. Esta secuencia de video fue llamada “Profesor_exterior”.

La escena donde se grabo esta secuencia de video, se observa que existen mas arboles que en la escena de la Figura 4.3, y el objetivo, es determinar si el algoritmo es robusto ante pequeños movimientos de las hojas de los arboles, considerando el movimiento de un solo objeto de interés y comprobar que el algoritmo es capaz de detectar el movimiento en trayectorias diagonales.

Las Figuras 4.5 a), b), c) y d) son la representación de un experimento donde dos alumnas se desplazan paralelamente al plano de la cámara, la alumna de blusa blanca se desplaza de izquierda a derecha en tanto que la alumna de blusa gris se desplaza de derecha a izquierda y ambas a una velocidad promedio de 0.83 m/s. Esta secuencia de video fue nombrada “Alumnas_2”.

El objetivo de este experimento es determinar la funcionalidad del **Algoritmo D** cuando existen dos objetos de interés en la escena procesada. Para realizar la detección de dos objetos de interés es necesario el uso del $umbral_{\text{área}}$ es decir, encontrar el objeto de mayor área relativa de movimiento y dado que $umbral_{\text{área}} = \text{área}_{\text{máxima}} \times 0.6$.

Si existe mas de un objeto moviéndose en la escena, y el área de los objetos es mayor o igual al $umbral_{\text{área}}$ podemos marcar mas de un objeto en movimiento.

Para validar el experimento, las alumnas se desplazan en dos planos transversales al foco de la cámara, haciendo que la distancia en los planos desplazamiento sea cercana, las áreas relativas de los objetos en movimiento tendrán tamaños similares, y por tanto el umbral podrá detectar ambas alumnas.

El experimento también permite validar la detección de movimiento con objetos que tienen trayectorias opuestas.

Las Figuras 4.6 a), b), c) y d) son la representación de un experimento donde tres alumnos se desplazan hacia la cámara. En estas tramas observamos un Alumno que se desplaza a una velocidad promedio de 3.3 m/s. También observamos dos alumnas que se desplazan a una velocidad promedio de 0.83 m/s. Todos los personajes se desplazan hacia la cámara. Esta secuencia de video fue nombrada "Alumnos_3".

Un objetivo de este experimento es determinar que el **Algoritmo D** es capaz de detectar el movimiento de objetos que se desplazan hacia la cámara y comprobar que la mayor área relativa de los objetos en movimiento define el $umbral_{\text{área}}$ y que al cambiar el tamaño de la mayor área relativa de los objetos en movimiento el $umbral_{\text{área}}$ cambia en relación al área máxima detectada.

De este modo otro objetivo del experimento es demostrar que el $umbral_{\text{área}}$ es adaptativo al tamaño del objeto y por tanto dependiendo el tamaño del área relativa de los objetos en movimiento, se pueden detectar más de un objeto en movimiento, siempre y cuando las áreas sean al menos el 60% del objeto en movimiento con la mayor área relativa.

Para validar el experimento se solicitó a los 3 alumnos iniciaran juntos sus trayectorias. Al alumno varón se le pidió que realizara su trayectoria corriendo en tanto que a las dos alumnas se les pidió que caminaran normalmente pero manteniendo la misma distancia. De este modo las alumnas se encuentran en el mismo plano transversal al foco de la cámara, y por tanto las áreas relativas de las alumnas en movimiento es similar.

En la Figura 4.6 a) se usó la trama 285 como trama actual y en el segundo reglón de imágenes observamos que la mayor parte de las ventanas de análisis consideradas movimiento, corresponden al alumno que realiza su desplazamiento corriendo. La mayor área relativa del objeto en movimiento es generada por que el alumno está más cerca a la lente de la cámara, ocupando más píxeles en la imagen que otros objetos, pero también puede generar más área debido a su rápido movimiento, pues los datos de las ventanas de análisis tienen mayores variaciones entre una trama y otra.

En este experimento también se concluyó que la velocidad del objeto no es un factor determinante para generar una mayor área relativa al objeto que se desplace a mayor velocidad, pues si el objeto se aleja del foco de la cámara, el número de píxeles necesarios para representar este objeto será menor, y puede existir otro objeto con una menor velocidad pero acercándose al foco de la cámara generando así una área relativa de movimiento mayor.

En la Figura 4.6 b) se usó la trama 310 como trama actual y en la Figura 4.6 c) la trama 330 es la trama actual. En ambas figuras observamos que el alumno que se encuentra corriendo es el objeto con mayor área relativa. Por tanto las ventanas que fueron consideradas movimiento y que fueron

agrupadas en objetos pero que tienen áreas menores al 60% del área mayor, son eliminadas y solo se considera el objeto de mayor área relativa.

En la Figura 4.6 d) la trama actual es la trama 510. Esta figura muestra a las dos alumnas que se desplazan en el mismo plano transversal al foco de la cámara, por esta razón el área relativa de ambos objetos en movimiento es similar y esta dentro del $umbral_{\text{área}}$ que se modificó en base al área mayor de esta escena, mostrando así que el umbral de área usado en el **Algoritmo D** es adaptativo.

4.5 RESULTADOS EXPERIMENTALES EN ESCENAS CON OBJETOS QUE SE DESPLAZAN A MAYOR VELOCIDAD QUE EL PROMEDIO DE LA VELOCIDAD DE DESPLAZAMIENTO EN HUMANOS.

Hasta el momento el **Algoritmo D** ha sido probado en escenarios exteriores e interiores y con uno, dos y tres objetos en la escena, La velocidad promedio de estos objetos corresponde a la velocidad promedio de las personas al caminar de 0.83 m/s, y la velocidad promedio de una persona al correr de 3.3 m/s.

En esta sección realizamos experimentos donde grabamos secuencias de video con objetos que se desplazan a unas velocidades mayores a los objetos de las secciones anteriores.

Las Figuras 4.7 a), b), c) y d) son la representación de un experimento donde un alumno se desplaza sobre una bicicleta en dirección a la cámara y con una velocidad promedio de 13 Km/H. En la Figura 4.7 a) se utiliza la trama 60, en la Figura 4.7 b) se utiliza la trama 72, en la Figura 4.7 c) la trama 92 y la Figura 4.7 d) la trama 113. Esta secuencia de video fue llamada “Ciclista”.

Las Figuras 4.8 a), b), c) y d) son la representación de un experimento donde un trabajador del IPN se desplaza sobre una cuatrimoto en dirección a la cámara y con una velocidad promedio de 10 Km/H. En la Figura 4.8 a) se utiliza la trama 380, en la Figura 4.8 b) se utiliza la trama 395, en la Figura 4.8 c) la trama 460 y la Figura 4.8 d) la trama 501. Esta secuencia de video fue llamada “Motoneta”.

La escena donde se grabaron las secuencias de video usadas para los experimentos presentados en las Figuras 4.7 y 4.8 no muestran variaciones considerables de las hojas de los árboles y por ello fueron los primeros experimentos realizados.

Las Figuras 4.9 a), b), c) y d) son la representación de un experimento donde se desplazan dos trabajadores abordo de un pequeño vehículo en dirección a la cámara y con una velocidad promedio de 15 Km/H. En la Figura 4.9 a) se utiliza la trama 340, en la Figura 4.9 b) se utiliza la trama 365, en la Figura 4.9 c) la trama 490 y la Figura 4.9 d) la trama 446. Esta secuencia de video fue llamada “Vehículo”.

La escena de la Figura 4.9 presenta mayor movimiento de los árboles que se encuentran en el fondo, al superar el movimiento no significativo de la secuencia de video, validamos la detección

de movimiento de objetos que se desplazan a mayor velocidad que el promedio en seres humanos y en escenas que presentan movimientos no significativos.

4.6 RESULTADOS EXPERIMENTALES EN ESCENARIOS NO CONTROLADOS

Hasta esta sección los experimentos se realizaron en escenas semi-controladas, limitando el número de personas, la dirección y la complejidad de las escenas al ubicar más o menos arboles.

En esta sección realizamos experimentos en escenarios cotidianos permitiendo el flujo libre de personas y objetos.

Las Figuras 4.10 a), b), c), d). Son la representación de un experimento donde se considero un pasillo de acceso libre, personas se encuentran desplazándose hacia la cámara y otras alejándose de ella, además en la escena se encuentra un vehículo moviéndose hacia la cámara. Esta secuencia de video fue llamada “Pasillo_principal”.

Este experimento permite validar la robustez del **Algoritmo D**, al evaluarlo en condiciones cotidianas de un pasillo donde existen movimientos que vienen hacia la cámara combinados con objetos en movimiento que se alejan de la cámara y la combinación de objetos que se desplazan a mayores velocidades de las promediadas en seres humanos.

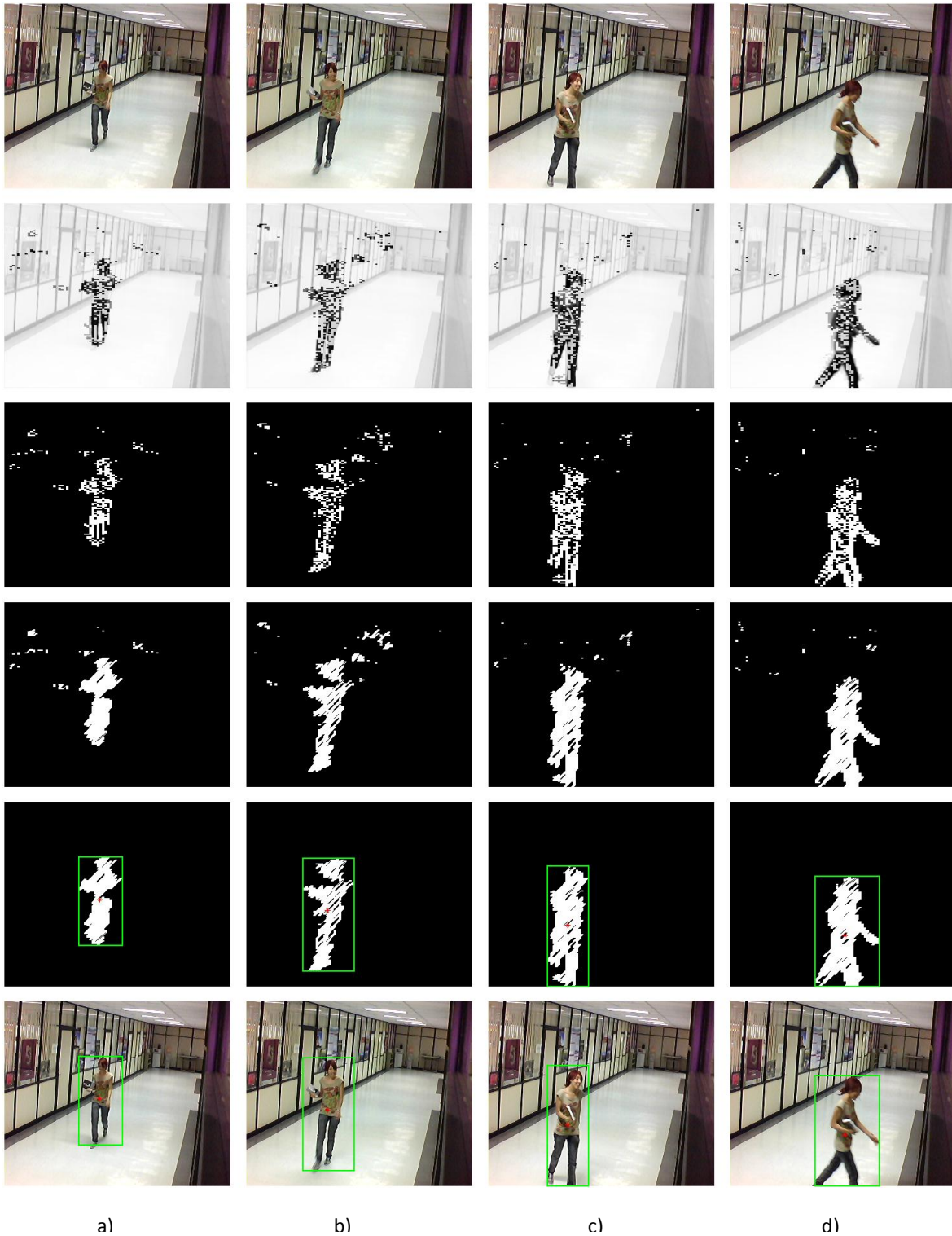


Figura 4. 1 Resultados del Algoritmo D aplicado a la secuencia de video "Laura_pasillo", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 180, b) Trama 205, c) Trama 230, d) Trama 250.

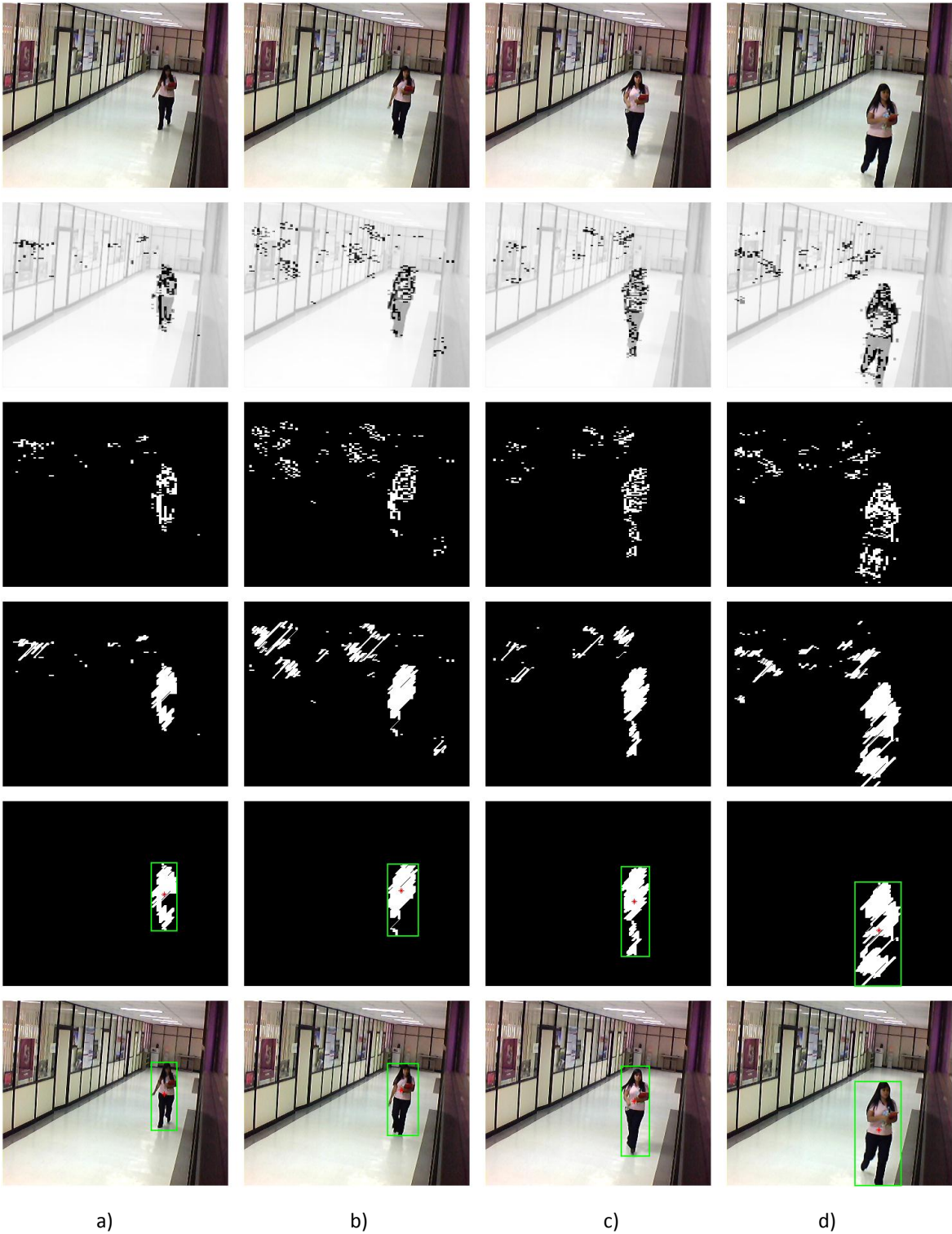


Figura 4. 2 Resultados del Algoritmo D aplicado a la secuencia de video "Mariana_pasillo", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 115, b) Trama 135, c) Trama 157, d) Trama 191.

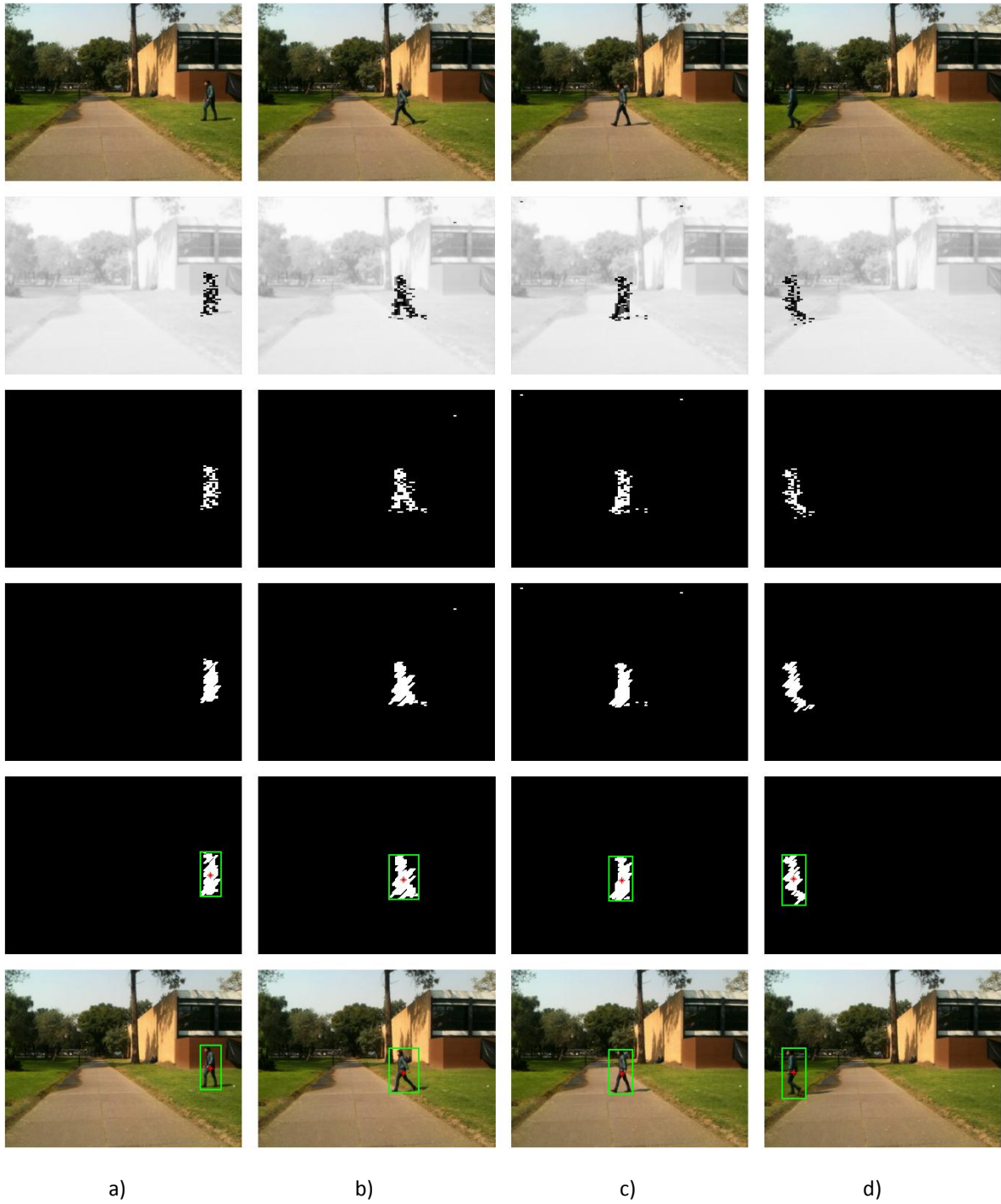


Figura 4. 3 Resultados del Algoritmo D aplicado a la secuencia de video "Laura_exterior", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 80, b) Trama 133, c) Trama 160, d) Trama 225.



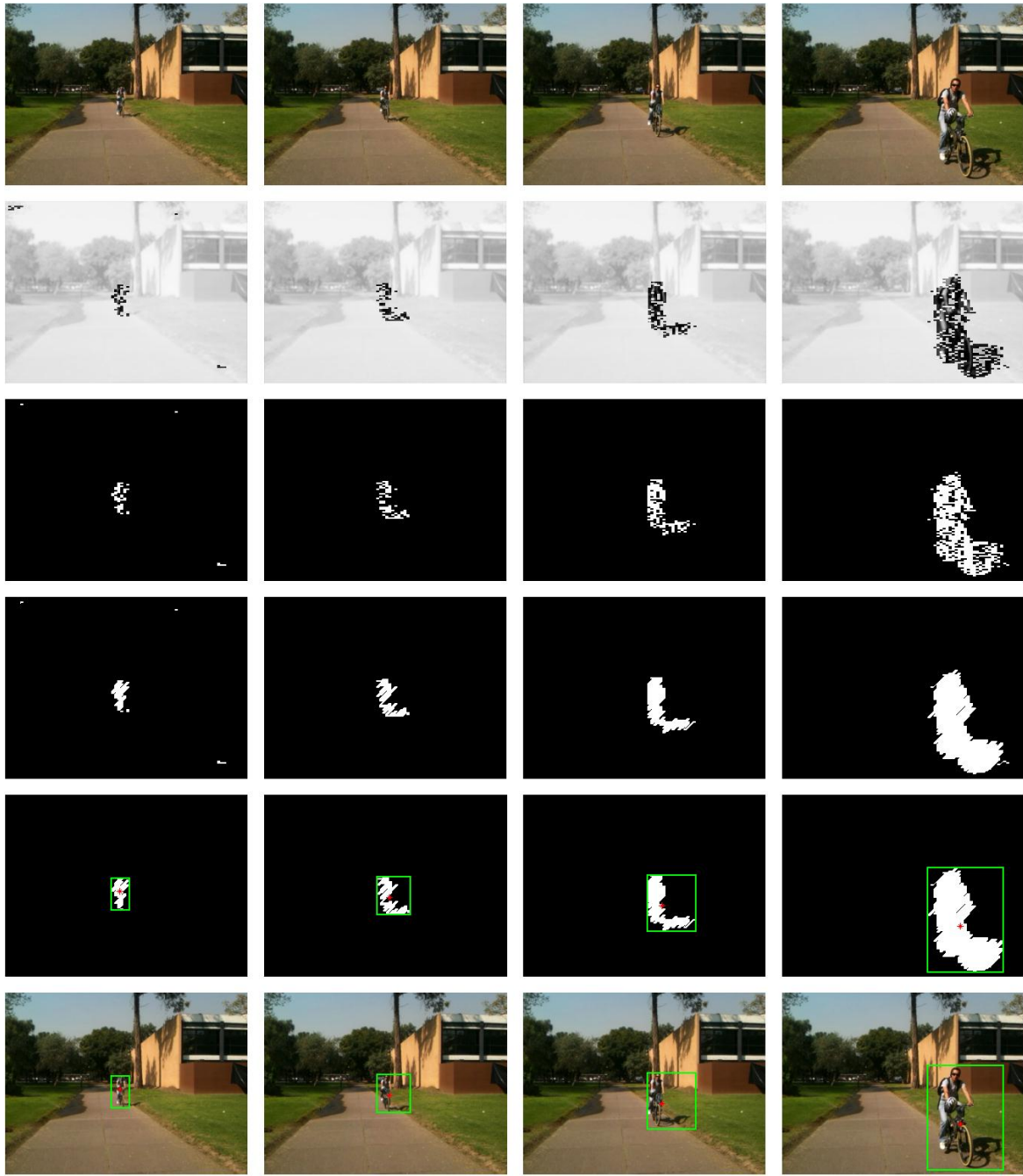
Figura 4. 4 Resultados del Algoritmo D aplicado a la secuencia de video "Profesor_exterior", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 67, b) Trama 98, c) Trama 121, d) Trama 151.



Figura 4. 5 Resultados del *Algoritmo D* aplicado a la secuencia de video "Alumnas_2", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 113, b) Trama 139, c) Trama 176, d) Trama 193.



Figura 4. 6 Resultados del Algoritmo D aplicado a la secuencia de video "Alumnos_3", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 285, b) Trama 310, c) Trama 330, d) Trama 510.



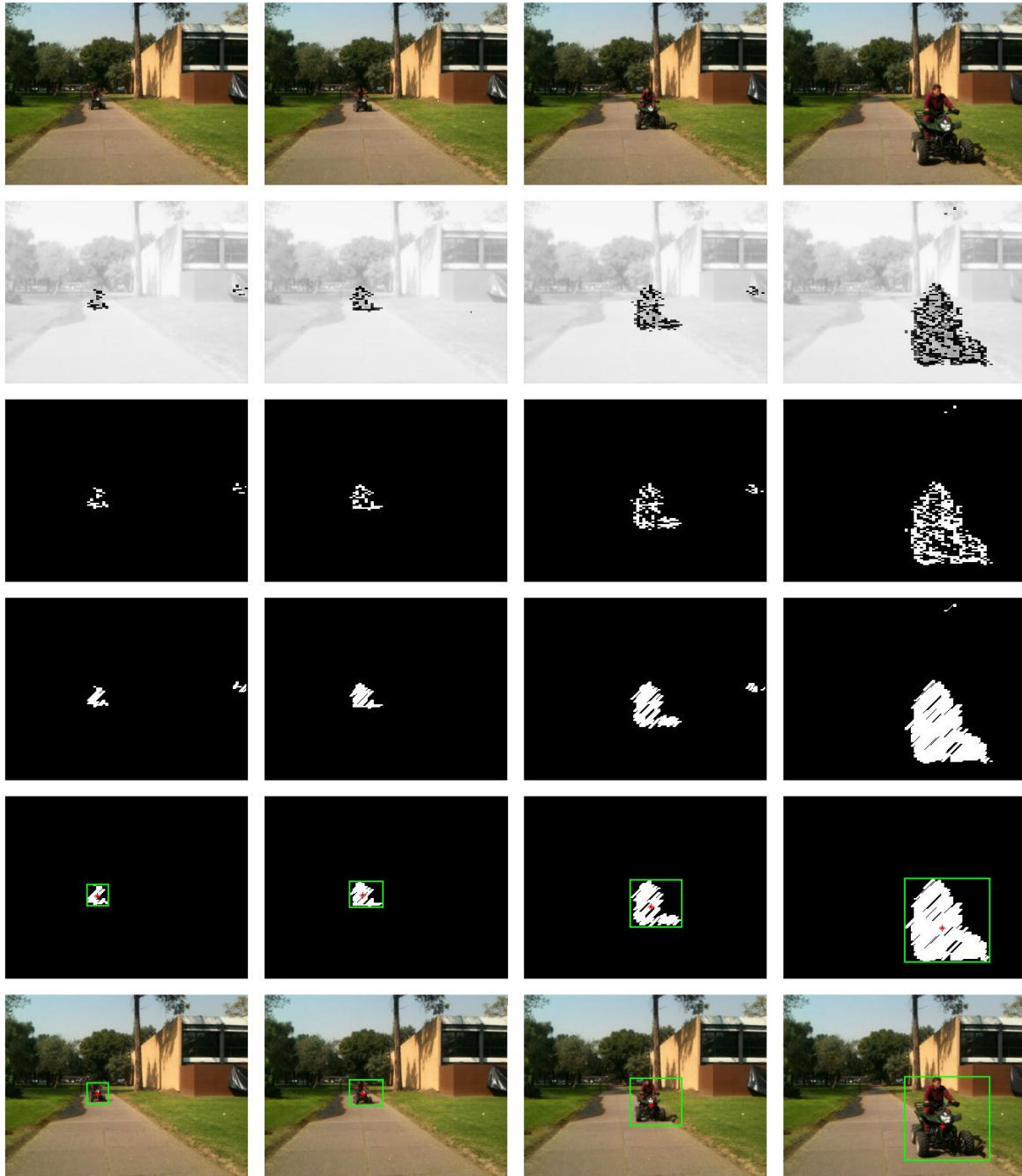
a)

b)

c)

d)

Figura 4. 7 Resultados del Algoritmo D aplicado a la secuencia de video "Ciclista", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 60, b) Trama 72, c) Trama 92, d) Trama 113.



a)

b)

c)

d)

Figura 4. 8 Resultados del Algoritmo D aplicado a la secuencia de video “Motoneta”, de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 380, b) Trama 395, c) Trama 460, d) Trama 501.

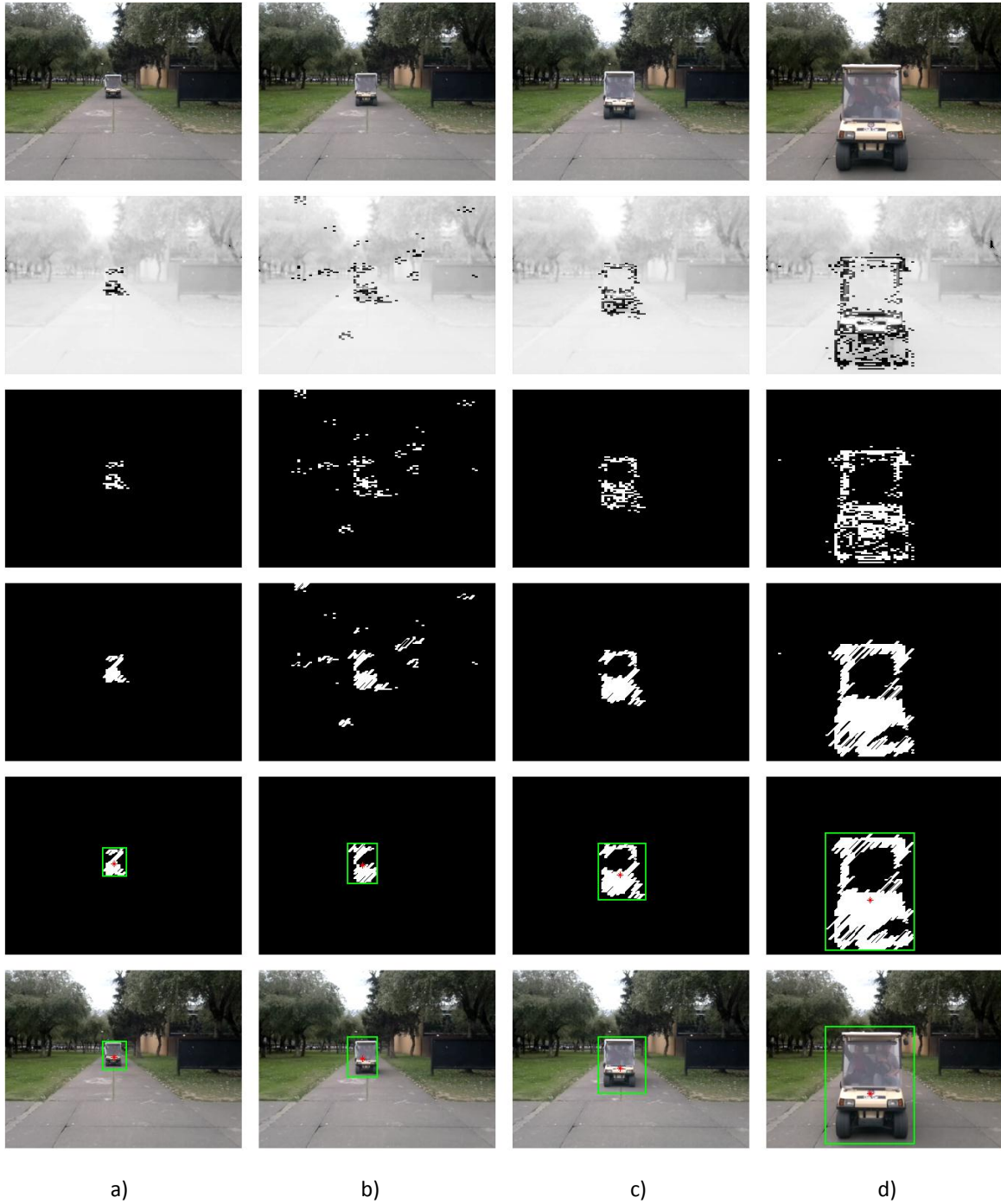


Figura 4. 9 Resultados del Algoritmo D aplicado a la secuencia de video "Vehículo", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 340, b) Trama 365, c) Trama 390, d) Trama 446.

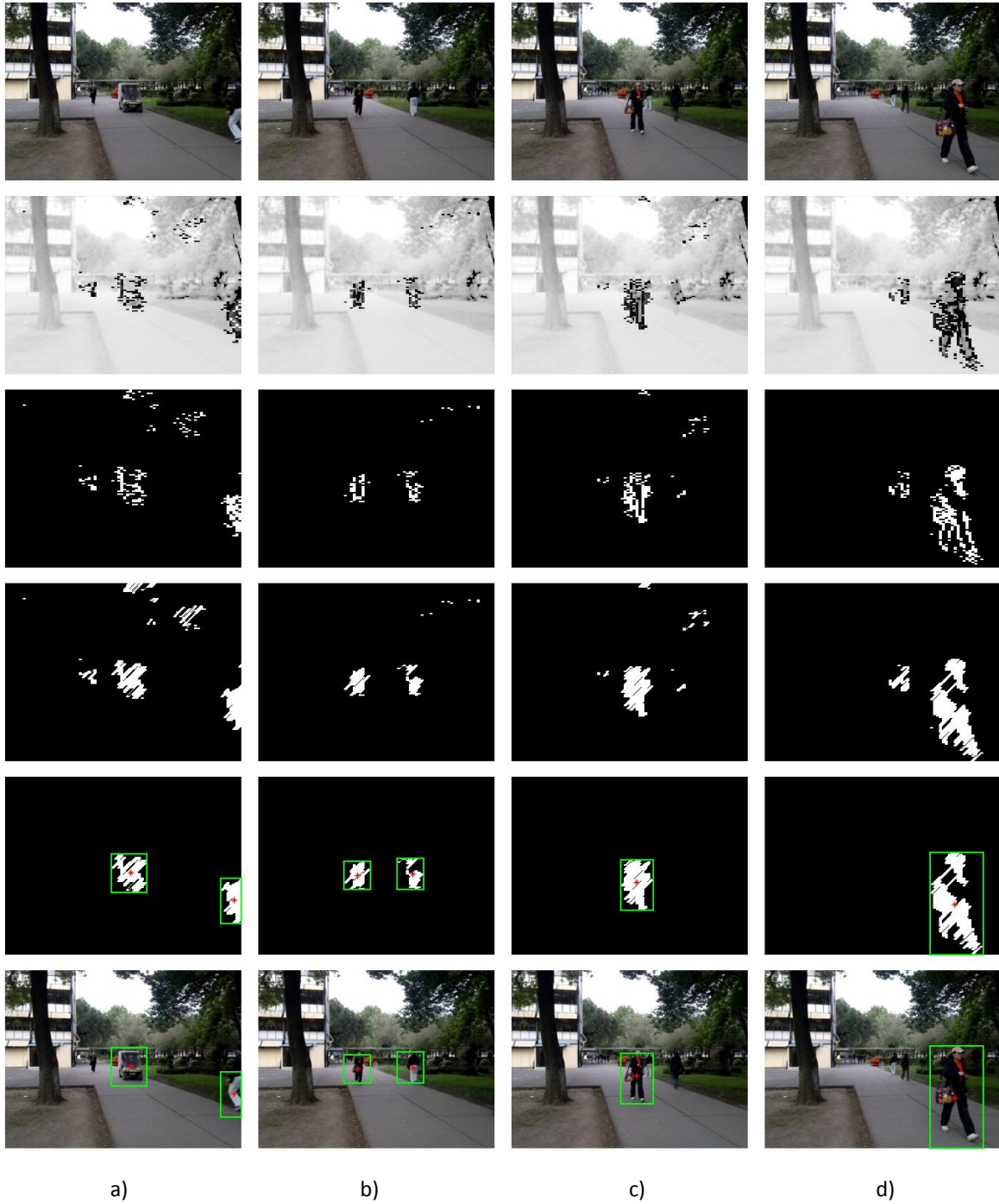


Figura 4. 10 Resultados del Algoritmo D aplicado a la secuencia de video "Pasillo_principal", de arriba abajo, Trama actual en espacio RGB, modificación del contraste en ventanas consideradas como movimiento, imagen binaria de ventanas consideradas movimiento, aplicación de la operación morfológica de cerradura, aplicación del umbral de área máxima, enmarcado y centroide del objeto en la trama actual en espacio RGB. a) Trama 140, b) Trama 290, c) Trama 375, d) Trama 438.

4.7 CONCLUSIONES

El algoritmo propuesto para detectar el movimiento en secuencias de video basado en el cálculo de la MAD de las ventanas de análisis, es efectivo en ambientes interiores y exteriores, con una o más personas, con objetos que son más grandes y con diferentes velocidades de movimiento de los objetos.

El algoritmo usa un umbral para el valor de la resta de las MAD lo que disminuye los efectos de cambios de iluminación.

Las operaciones de pre-procesamiento permitieron eliminar pequeños movimientos contenidos en los 4 bits menos significativos y además difuminar los bordes de los objetos para una mejor detección de movimiento.

El algoritmo propuesto le da prioridad al objeto de mayor área de movimiento, si existe más de un objeto, el objeto de mayor área de movimiento proporciona un valor que cambia dinámicamente con la escena, con este valor del área de movimiento generamos un umbral de área que también es dinámico.

Al reducir las ventanas de análisis a 2x4 píxeles, las imágenes binarias obtenidas delimitan la silueta de los objetos en movimiento, métodos propuestos por otros autores utilizan técnicas de procesamiento especializadas en la segmentación de imágenes, para obtener la silueta del objeto en movimiento, de este modo el algoritmo propuesto obtiene la silueta del objeto en movimiento sin utilizar técnicas de segmentación.

4.8 TRABAJO A FUTURO

En el sistema de vigilancia visual, mostrado en la Figura 2.1 del capítulo 2, se observa que una vez que se detecta movimiento significativo, el siguiente proceso es la clasificación de los objetos en movimiento y determinar si se trata de un vehículo, una persona o un grupo de personas, por lo que como trabajo a futuro se procede a la clasificación de los objetos en movimiento.

Las etapas de este proceso consisten en implementar técnicas para eliminar la oclusión de objetos en la escena, pues con esto sabremos realmente cuantos objetos se encuentran en la escena.

Al tener el número de los objetos en la escena se realiza una comparativa de diversas características de la imagen y el movimiento para determinar el tipo de objeto que se está moviendo, por ejemplo una persona, un vehículo o una mascota, etc.

4.4 REFERENCIAS BIBLIOGRÁFICAS

[1]Portillo-Portillo, J.; Velázquez-González, J. S.; Gallegos-Funes, F.J.; Rosales-Silva, A.J."Detección de Movimiento en secuencias de video" *Participación de la Mujer en la Ciencia, CIO, Mayo 2011, León Guanajuato.*

[2]Portillo-Portillo, J.; Gallegos-Funes, F.J.; Rosales-Silva, A.J, "Movement Detection using an Order Statistics Algorithm" *Proc. SPIE 8437, Real-Time Image and Video Processing 2012, Brussels, Belgium, 843700 (2012)*

CAPÍTULO 5 PARTICIPACIÓN EN CONGRESOS

5.1 CONGRESOS NACIONALES

Portillo-Portillo, J.; Velázquez-González, J. S.; Gallegos-Funes, F.J.; Rosales-Silva, A.J. "Detección de Movimiento en secuencias de video" *Participación de la Mujer en la Ciencia, CIO, Mayo 2011, León Guanajuato.*

5.1 CONGRESOS INTERNACIONALES

Portillo-Portillo, J.; Gallegos-Funes, F.J.; Rosales-Silva, A.J, "Movement Detection using an Order Statistics Algorithm" *Proc. SPIE 8437, Real-Time Image and Video Processing 2012, Brussels, Belgium, 84370O (2012)*