UNIVERSIDAD DE GUANAJUATO

CAMPUS IRAPUATO - SALAMANCA
DIVISIÓN DE INGENIERÍAS

Seguimiento continuo de objetos en una red de cámaras

TESIS

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN INGENIERÍA ELÉCTRICA
(Opción: Instrumentación y Sistemas Digitales)

PRESENTA:

Ing. Lubín Enrique Rincón Chacón

DIRECTOR:

Dr. Víctor Ayala Ramírez

SALAMANCA, GTO Mayo, 2018

Dedicatoria

Dedicado a
mi familia

Agradecimientos

Agradezco profundamente a mis padres, de los cuales siempre estaré orgulloso y agradecido. Por su apoyo incondicional y por siempre motivarme a lograr mis metas.

A mi novia Andrea y su familia, por tan gran apoyo y motivarme a seguir mis sueños. Gracias.

A mi asesor, Dr. Víctor Ayala Ramírez por permitirme ser parte de su equipo de trabajo. Por su confianza y valiosas enseñanzas durante mi formación académica. Así mismo, agradezco a los Doctores del laboratorio por su apoyo.

A mis compañeros del LaViRIA que siempre estuvieron ahí, por su apoyo y amistad, gracias.

A mis amigos, compañeros y profesores con los que tuve la oportunidad de convivir durante mi estancia en la carrera.

Agradecimientos Institucionales

A la División de Ingenierías Campus Irapuato-Salamanca de la Universidad de Guanajuato, por la formación académica recibida a través del programa de "Maestría en Ingeniería Eléctrica: Instrumentación y Sistemas Digitales".

Universidad de Guanajuato

Al Laboratorio de Visión, Robótica e Inteligencia Artificial (LaViRIA), por la formación integral recibida durante mi estancia en él, y por el espacio y los recursos proporcionados para la realización de este proyecto.

Perception, cognition and robotics sinergy

Al Consejo Nacional de Ciencias y Tecnologías (CONACYT) por el apoyo financiero mediante la beca con registro (CVU/Becario) 736551/598646.

CONACYT

Índice general

1. Introducción **1**
1.1. Objetivo General 2
1.2. Objetivos Específicos 2
1.3. Justificación 2
1.4. Plan del Trabajo 3

2. Formulación del Problema **4**
2.1. Materiales y Métodos 5
2.2. Trabajos Relacionados 8
2.3. Descripción General del Sistema Propuesto 9
2.4. Conclusiones del Capítulo 9

3. Seguimiento Continuo de Objetos en una Red de Cámaras **11**
3.1. Descripción del Sistema Propuesto 12
3.2. Seguimiento Visual de Objetos 14
3.2.1. Problemas asociados a la adquisición de imagen 14
3.2.2. Modelo del objeto 15
3.2.3. Estrategias de búsqueda e identificación del modelo en la imagen 16
3.2.4. Estrategias de actualización del modelo 17
3.2.5. Estrategia de predicción de la dinámica del objeto 18
3.3. Fusión de Información 22
3.3.1. Agregación de la información disponible 23
3.3.2. Selección de información 28
3.4. Conclusiones del Capítulo 31

4. Pruebas y Resultados **32**
4.1. Protocolo de Pruebas 33
4.2. Métricas de Evaluación 33
4.3. Pruebas en Vídeos 35
4.3.1. Seguimiento Visual de Objetos 35
4.3.2. Fusión de Información 37

4.4. Comparación con Otros Métodos 44
4.5. Conclusiones del Capítulo 45

5. Conclusiones y Perspectivas **46**

Anexos **48**

A. Seguimiento Visual de Objetos **49**

B. Agregación de la Información Disponible **53**

C. Selección de Información **57**

Bibliografía **62**

Índice de figuras

Capítulo II

2.1. Modelo de cámara Pinhole [15]. 5
2.2. Transformación euclidiana entre el mundo y el centro de la cámara [15]. 6
2.3. Correspondencia entre líneas de campo de visión (FOV) [18]. . . . 9

Capítulo III

3.1. Clasificación de los métodos de seguimiento de objetos [37]. 11
3.2. Sistema propuesto para el seguimiento continuo de objetos en una red de cámaras. 13
3.3. Ejemplo de la imagen de salida del sistema propuesto. Aquí, la imagen de mayor tamaño representa la cámara con mejor vista del objeto y las imágenes de menor tamaño representan todas las cámaras del sistema. 13
3.4. Sistema de seguimiento visual. 14
3.5. Clasificación de los métodos de modelado de objetos [37]. 16
3.6. Muestra del color objetivo (recuadro en la esfera). 16
3.7. Desplazamiento de una función de membresía. 18
3.8. Zona de búsqueda sin predicción. 19
3.9. Zona de búsqueda con predicción. 19
3.10. Operación completa del filtro de Kalman. 20
3.11. Sistema de fusión de información. 23
3.12. Sistema para la agregación de la información disponible. 23
3.13. Región utilizada por la geometría epipolar. 24
3.14. Selección de los puntos utilizados por la geometría epipolar. . . . 25
3.15. Zona de búsqueda generada por la geometría epipolar. 26
3.16. Zona de búsqueda final. 27
3.17. Proceso de correspondencia de los centroides en dos cámaras. . . . 29

Capítulo IV

4.1. Evaluación de los píxeles encontrados por el sistema comparado con los píxeles de la verdad de referencia. 34
4.2. Distribución espacial de las cámaras en el escenario del tercer vídeo. 41
4.3. Comportamiento del error global para cada valor de la constante de proporción, por cada sujeto de prueba. 42
4.4. Selección automática de la cámara con mejor vista del objeto. . . 43

ANEXOS

A.1. Seguimiento visual de una taza azul en un ambiente semi-controlado. 50
A.2. Gráficas entre la verdad de referencia y los valores calculados por el sistema para cada cuadro del vídeo uno. 51
A.3. Gráficas de la variación dinámica de las componentes del espacio de color CIELab para cada cuadro del vídeo uno. 52

B.1. Seguimiento visual de un suéter azul en un ambiente no controlado visto desde la cámara 1 en el vídeo dos. 54
B.2. Seguimiento visual de un suéter azul en un ambiente no controlado visto desde la cámara 2 en el vídeo dos. 55
B.3. Seguimiento visual de un suéter azul en un ambiente no controlado visto desde la cámara 3 en el vídeo dos. 56

C.1. Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara 1 en el vídeo tres. 58
C.2. Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara 2 en el vídeo tres. 59
C.3. Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara 3 en el vídeo tres. 60
C.4. Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara con mejor vista del objeto en el vídeo tres. 61

Índice de tablas

Capítulo IV

4.1. Mejores resultados del análisis del primer vídeo. 36
4.2. Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el primer vídeo. . . 37
4.3. Mejores resultados del análisis del segundo vídeo para la primera cámara. 38
4.4. Mejores resultados del análisis del segundo vídeo para la segunda cámara. 39
4.5. Mejores resultados del análisis del segundo vídeo para la tercera cámara. 39
4.6. Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el segundo vídeo en la cámara 1. 39
4.7. Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el segundo vídeo en la cámara 2. 40
4.8. Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el segundo vídeo en la cámara 3. 40
4.9. Análisis de la frecuencia de operación del segundo vídeo. 40
4.10. Mejores resultados del análisis del tercer vídeo para la cámara seleccionada automáticamente. 43
4.11. Análisis de la desviación estándar del error obtenido del tercer vídeo en la cámara con mejor vista del objeto. 44
4.12. Comparación del estado del arte y el sistema desarrollado para el primer vídeo analizado. 45

Capítulo 1

Introducción

El seguimiento de objetos mediante múltiples cámaras es interesante porque permite identificar la trayectoria que ha seguido un objeto mientras sea visible por alguna de las cámaras disponibles. Por ejemplo, en una aplicación de video-vigilancia, puede ser interesante saber las trayectorias de algunos de los elementos que conforman la escena. Un segundo ejemplo puede ser la verificación de la realización de algunas operaciones en un proceso de manufactura industrial.

Al comparar los dos ejemplos anteriores podemos notar que los sistemas de seguimiento de objetos requieren:

- La definición de modelos para los objetos de interés así como de sistemas de referencia con respecto a los cuales los objetos son localizados.
- Tomar en cuenta el contexto de operación que complica de manera importante el procesamiento de la información visual adquirida por las cámaras. Por ejemplo, los cambios de iluminación, la aparición de sombras y oclusiones, así como las especificaciones particulares de cada uno de los componentes incluidos en el sistema.
- Determinar según la aplicación, si lo que se requiere es la información agregada de todas las cámaras del sistema, o si se requiere definir un criterio de utilidad, que deba ser optimizado para la selección de la información proveniente de uno o varios de los sensores visuales.

Este trabajo tiene la finalidad de estudiar este problema para lograr la implementación de un sistema útil para el seguimiento visual de objetos en contextos de aplicaciones diversas. Este propósito requiere que se describan las elecciones de diseño usadas para construir el sistema y que se refieran las pruebas que validan el tomar tales decisiones.

1.1. Objetivo General

El objetivo de esta tesis es el desarrollo de un sistema de seguimiento continuo de objetos en múltiples cámaras, con la finalidad de determinar e indicar la ubicación de los objetos en cuadros consecutivos.

1.2. Objetivos Específicos

Los objetivos esperados del trabajo de tesis son los siguientes:

- Estudio de técnicas para el seguimiento visual de objetos.
- Desarrollo de al menos una variante para el seguimiento continuo de objetos en una red de cámaras, en comparación con alguna técnica del estado del arte, para el seguimiento de objetos en múltiples cámaras.
- Validación de esta técnica mediante la evaluación de su desempeño contra una técnica del estado del arte para el seguimiento de objetos en múltiples cámaras.
- Documentación del proceso de desarrollo del proyecto.

1.3. Justificación

El seguimiento de objetos basado en video es el proceso de encontrar las regiones donde están ubicados los objetos en cuadros consecutivos [4], siendo una desafiante y atractiva área de investigación en la visión por computadora. La tecnología de seguimiento de objetos puede ser utilizada ampliamente en vídeo vigilancia, transporte inteligente y robots inteligentes. Sin embargo, debido a la limitación del campo de visión de una sola cámara, es muy difícil seguir varios objetos al mismo tiempo [6].

La video vigilancia inteligente juega un papel cada vez más importante en la sociedad actual para la seguridad de las áreas públicas. Z. Chen *et al.* [9], definen que un sistema de video vigilancia consiste en una red de cámaras con campo de visión compartida o no compartida. Y. Cai *et al.* [5], señalan que en lugar de tener una cámara de alta resolución con un campo de visión limitado, múltiples cámaras proporcionan una solución para la video vigilancia al extender el campo de visión de una sola cámara. Esto trae consigo retos y oportunidades en la detección y seguimiento de objetos.

Las técnicas de seguimiento visual de objetos mediante múltiples cámaras, buscan identificar la trayectoria de los objetos en diferentes escenarios de aplicación. Y. Chen *et al.* [8], indican que el rendimiento del sistema de seguimiento de objetos puede ser degradado debido a interferencias visuales, tales

como variaciones de iluminación, cambios de fondo, deformación no rígida de los objetos y oclusión parcial, entre otros.

Considerando lo anterior, el desarrollo de métodos efectivos y eficientes en el seguimiento de objetos en múltiples cámaras sigue teniendo relevancia. En particular, esta tesis explorará alternativas para realizar el seguimiento continuo de objetos en una red de cámaras tomando en cuenta el consumo de recursos computacionales.

1.4. Plan del Trabajo

El presente trabajo de tesis está distribuido en 5 capítulos de la siguiente manera:
En el Capítulo 2, se realiza la descripción y exploración del problema a resolver. Además, se muestran algunos métodos en el estado del arte que buscan resolver dicho problema.
En el Capítulo 3, se identifica el alcance y se describe la metodología propuesta para dar una solución a esta problemática.
En el Capítulo 4, es presentado el protocolo de pruebas utilizado para la evaluación del desempeño del sistema propuesto. Adicionalmente, se muestran los resultados experimentales obtenidos.
En el Capítulo 5, se presentan las conclusiones y perspectivas de este trabajo.

Capítulo 2

Formulación del Problema

Los sistemas de seguimiento de objetos con una o mas cámaras buscan identificar la trayectoria que ha seguido un objeto de manera continua utilizando todas las cámaras o vistas disponibles. Sin embargo, esta estimación puede ser afectada negativamente por diversos factores, estos pueden ser: cambios de iluminación, oclusiones, aparición de sombras y alta velocidad del objeto. Considerando lo anterior, se puede intuir que el seguimiento continuo de objetos en una o mas cámaras es una tarea que, dependiendo del escenario, puede llegar a ser muy compleja.

Muchos sistemas de seguimiento continuo de objetos con una cámara se ocupan de las oclusiones [7][26][27][32]. Estos métodos pueden manejar la oclusión hasta cierto punto, pero se vuelven menos eficaces cuando los objetos sufren oclusiones completas a largo plazo. Además, estos sistemas están limitados en el alcance de sus aplicaciones, ya que, incluso aplicaciones simples de seguimiento visual de objetos exigen el uso de múltiples cámaras [18].

El seguimiento de objetos utilizando múltiples cámaras ha atraído un creciente interés en la investigación en los últimos años, en gran medida impulsado por su amplia cobertura espacial lo que es ventajoso en escenarios complejos [38]. Tradicionalmente, muchos de estos sistema modelan los objetos con una alta dimensión de información, pudiendo así, codificar efectivamente los datos de entrada con más información. Sin embargo, la computación precisa de esta información se vuelve intratable e imposible para un sistema de seguimiento en tiempo real debido al costo computacional y los requisitos de memoria [30]. Además, muchos de estos sistemas realizan el seguimiento de objetos sin considerar cuál cámara puede ofrecer una mejor vista del objeto.

En este capítulo se definirán conceptos importantes utilizados en el seguimiento de objetos. Adicionalmente, se mencionarán algunos de los métodos utilizados en la literatura y se mostrará una breve descripción del sistema propuesto.

2.1. Materiales y Métodos

La región objetivo, el modelado de cámaras, la información de apariencia e información de movimiento son conceptos importantes para este trabajo. En esta sección se hará una descripción de dichos elementos.

Región Objetivo

La región objetivo es la representación del objeto u objetivo en los sistemas de seguimiento visual, el objeto suele estar representado por un cuadro delimitador, como en [5] y en muchos otros algoritmos por una elipse, como en [11]. El argumento común en contra de un cuadro delimitador es que los píxeles de fondo pueden confundirse con los píxeles perteneciente al objeto. Sin embargo, la principal ventaja de los cuadros delimitadores es el bajo número de parámetros asociados para representar el área del objeto [31].

Modelado de Cámaras

Las cámaras permiten hacer la proyección entre un mundo 3D (espacio de objetos) a imágenes 2D [15]. Estas proyecciones son representadas a través de modelos matemáticos.

Existe una variedad de modelos matemáticos que describen las proyecciones de las cámaras. Sin embargo, la mayoría de las cámaras usan el modelo de cámara sencillo y más especializado, siendo éste el modelo de cámara pinhole (o por su nombre en español, estenopeica) [15]. La descripción gráfica de este modelo se evidencia en la Figura 2.1.

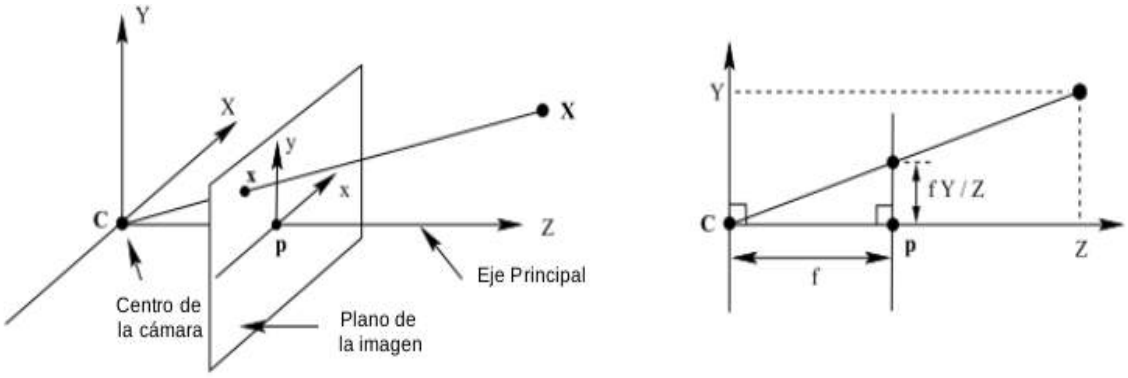

Figura 2.1: *Modelo de cámara Pinhole [15].*

En la figura, C es el centro de la cámara y P el punto principal. La distancia o longitud focal f, representa la distancia entre el centro de la cámara y el plano de la imagen.

El modelo de cámara pinhole puede ser representado matemáticamente por medio de dos matrices. Estas matrices son: la matriz de calibración intrínseca y la matriz de calibración extrínseca.

La matriz de calibración intrínseca en coordenadas homogéneas observada en la Ecuación 2.1, permite proyectar puntos 3D cuyo origen de coordenadas se encuentra en el centro de la cámara, a puntos 2D en el plano de la imagen.

$$K = \begin{bmatrix} f_x & 0 & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.1}$$

donde, x_0 y y_0 son las coordenadas x e y del centro del plano de la imagen. f_x y f_y representan la longitud focal de la cámara en términos de dimensiones de distancia en la dirección x e y, respectivamente.

Por otra parte, la matriz de calibración extrínseca representa la rotación y traslación de la cámara respecto a algún marco de coordenadas. En general, los puntos en el espacio se expresarán en términos de un marco de coordenadas euclidiano diferente, conocido como el marco de coordenadas del mundo. Por lo que, el marco de coordenadas del centro de la cámara "C" y el marco de coordenadas del mundo "O" están relacionadas mediante una rotación y una traslación (ver Figura 2.2). Esta matriz se puede observar en la Ecuación 2.2.

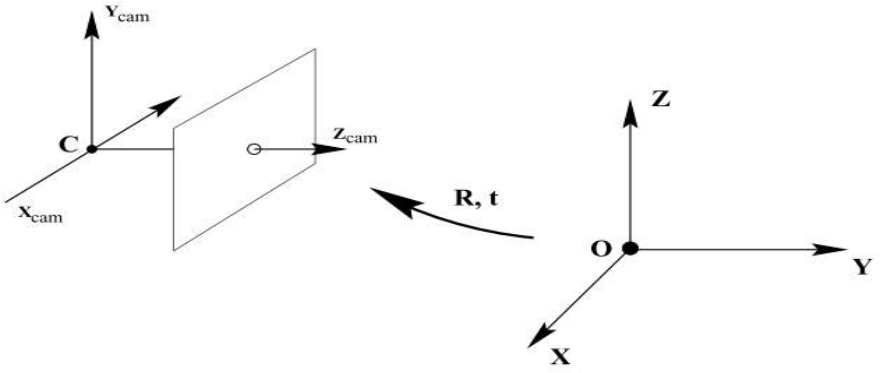

Figura 2.2: *Transformación euclidiana entre el mundo y el centro de la cámara [15].*

$$\begin{bmatrix} R & | & t \end{bmatrix} = \left[\begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_2 \end{array}\right] \tag{2.2}$$

donde, R es la matriz de rotación y t el vector de traslación que relacionan los dos marcos de coordenadas.

Finalmente, el modelado de las cámaras consiste esencialmente en obtener los parámetros de la matriz de calibración intrínseca y extrínseca. Con este modelo, cualquier punto 3D en el marco de coordenadas del mundo puede ser mapeado al plano de la imagen de la cámara, utilizando la Ecuación 2.3.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K * \begin{bmatrix} R & |t \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \left[\begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_2 \end{array}\right] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.3}$$

Aquí, x y y representan las coordenadas x e y de un punto en el plano de la imagen. X, Y y Z son las coordenadas de un punto 3D en el marco de coordenadas del mundo. Las ecuaciones presentadas en esta sección pueden ser revisadas en [15].

Información de Apariencia

La información de apariencia del objeto está representada por características visuales. Esto implica que una determinada propiedad es visualmente constante en cuadros consecutivos. Estas propiedades pueden ser el color, la textura y la forma del objeto [31].

Modelos de Color

Una de las propiedades mas utilizadas en el seguimiento de objetos mediante múltiples cámaras es el color [5][16][38]. Esto se debe a que las señales de color son invariantes a la rotación y escala, y requieren menos tiempo de cómputo que otras propiedades. Sin embargo, las señales de color son sensibles a los cambios de iluminación, incluso bajo condiciones de iluminación controlada, el color de un objeto puede variar, ya que su movimiento ocasiona que el objeto refleje la luz de diferente manera [39].

En los sistemas donde el color es utilizado, debe elegirse algún modelo para representar dicho color. Existen varios modelos de color como el RGB, HSI, YIQ, YUV y CIELab que han resultado adecuados para diferentes aplicaciones, como seguimiento de objetos, reconocimiento de caras y segmentación. Los modelos de color CIELab y HSI son unas de las formas de representar el color mas usadas en aplicaciones de seguimiento de objetos [24], llamados también modelos de color perceptual debido a que tratan de describir el color como lo hace el sistema de visión humano. Además, estos modelos separan las componentes de color de la componente de iluminación.

Información de Movimiento

En muchos sistemas de seguimiento donde un objeto se está moviendo, se asume que el objeto está cerca de la posición anterior, y éste puede encontrarse mediante una búsqueda uniforme alrededor de dicha posición [31]. Este movimiento del objeto permite obtener información espacial y temporal que puede ser utilizada para estimar la trayectoria y posición de los objetos en el escenario.

2.2. Trabajos Relacionados

Algunos enfoques tratan de utilizar la información de apariencia como característica de información principal, y modelos estadísticos para la correspondencia de los objetivos entre las diferentes cámaras o vistas disponibles. Y. Cai *et al.* [5] proponen el seguimiento de peatones en un escenario donde el campo de visión de las cámaras no es compartido, utilizando cuatro histogramas de color para representar los peatones en cada cámara. La correspondencia de los objetivos en cada cámara es realizada utilizando un modelo bayesiano.

H. Hsu *et al.* [16] presentan un sistema de seguimiento de personas a través de múltiples cámaras, donde, se utiliza el color y un marco bayesiano para la clasificación del fondo y del primer plano. Posteriormente, cuando la persona es segmentada aplican la operación de etiquetado de componente de conexión para identificar los objetos en movimiento. Estos objetos son comparados e identificados en las cámaras por medio de sus histogramas de color. Finalmente, luego de la identificación de la persona, se utiliza el algoritmo mean shift para su seguimiento.

M. Li *et al.* [19] proponen un modelo espacio-temporal de mezcla de gausianas basado en las estadísticas de información de espacio y tiempo para el seguimiento de objetos entre múltiples cámaras. Este modelo pretende predecir las cámaras donde los peatones aparecerán, luego de que éstos, desaparezcan de la vista de otras cámaras.

Algunos métodos para el seguimiento de objetos en múltiples cámaras aprovechan las características geométricas de la posición de las cámaras y del escenario. En el método de Y. Yun *et al.* [38] se utiliza el color para representar los objetos y un modelo bayesiano para el seguimiento de éstos en cada cámara. Luego, se combinan las restricciones de la homografía y geometría epipolar para hacer la correlación de dichos objetos.

S. Khan y M. Shah *et al.* [18] plantean un sistema de seguimiento de personas en múltiples cámaras, sin calibrar y con campos de visión compartido. Este sistema se enfoca principalmente en la correspondencia de las personas en cada cámara. Esta correspondencia se logra calculando la distancia mínima entre los pies de las personas y las líneas de campo de visión (líneas FOV, por sus siglas en ingles). Estas líneas son esencialmente los bordes de la huella de una cámara vista en otras cámaras, como se observa en la Figura 2.3.

En el método propuesto por Z. Cai *et al.* [6] para el seguimiento de peatones en una sistema de múltiples cámaras, se adoptan los histogramas de gradientes orientados para la detección de los peatones en cada cámara. La correspondencia de dichos peatones se realiza mapeando sus pies de una vista a otra utilizando la homografía del plano del suelo, y transformando un punto de la cabeza de los peatones de una vista a una recta en otra vista, utilizando la geometría epipolar.

Yun *et al.* [39] plantean un sistema de seguimiento de manos en múltiples cámaras para procesos industriales. Inicialmente, la identificación de las manos se realiza utilizando histogramas de color y el algoritmo de agrupamiento K-medias

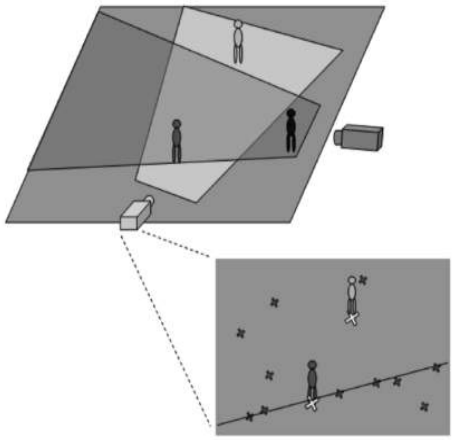

Figura 2.3: *Correspondencia entre líneas de campo de visión (FOV) [18].*

en cada cámara. La correspondencia de las manos en cada cámara se logra minimizando la distancia entre el centroide de cada mano y la línea proyectada del centroide de dicha mano en otra vista, utilizando la geometría epipolar.

2.3. Descripción General del Sistema Propuesto

El sistema propuesto busca resolver el problema de seguimiento continuo de objetos en una red de cámaras, en escenarios donde las cámaras están modeladas y comparten entre sí parte de su campo de visión.

Inicialmente, el objeto es modelado utilizando su información de apariencia a partir de la selección manual de una región del objeto en una cámara. Posteriormente, la información de apariencia obtenida es utilizada para modelar el objeto en las otras cámaras a través de un algoritmo de segmentación por regiones. Para ello, es encontrada la región que pertenece al objeto al combinar las restricciones geométricas del escenario y la información de movimiento de la escena. Una vez modelado el objeto es seguido de manera independiente en cada cámara estimando en cada iteración su ubicación y su región objetivo.

Finalmente, en este trabajo se propone un método para seleccionar en cada iteración la cámara que tiene mejor vista del objeto. Esencialmente, en esta propuesta se considera que la cámara con mejor vista es elegida por medio de una combinación de características, siendo estas, la cámara con la ubicación más cercana al objeto y la cámara que posee el mayor área visual del objeto.

2.4. Conclusiones del Capítulo

En este capítulo se presentó el problema del seguimiento continuo de objetos en una red de cámaras, así como la definición de algunos conceptos importantes

y un resumen de los trabajos usados en el estado del arte para la resolución de este problema. En esta investigación se observó que la información de apariencia, la información de movimiento, las restricciones geométricas y los modelos estadísticos son las características de información más utilizadas para resolver esta problemática. Por último, se realizó una breve descripción del sistema propuesto.

Capítulo 3

Seguimiento Continuo de Objetos en una Red de Cámaras

Como se mencionó en el capítulo anterior, los sistemas de seguimiento visual de objetos mediante múltiples cámaras permiten identificar la trayectoria que ha seguido un objeto mientras sea visible por alguna de las cámaras disponibles. Estos sistemas traen consigo retos y oportunidades en el desarrollo de métodos para su implementación.

En la literatura existen muchos métodos que buscan solucionar el problema del seguimiento visual de objetos. Dependiendo su aplicación, estos métodos pueden clasificarse en tres tipos: Seguimiento por puntos, núcleo y silueta (ver Figura 3.1) [37].

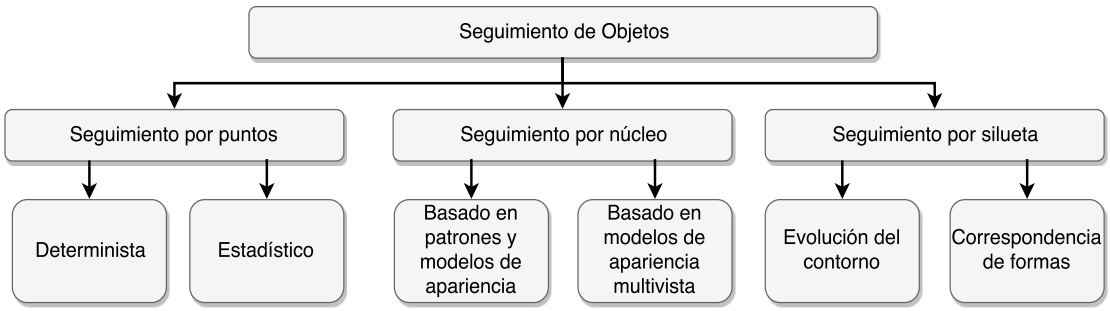

Figura 3.1: *Clasificación de los métodos de seguimiento de objetos [37].*

La selección del tipo de sistema de seguimiento debe realizarse cuidadosamente; una de las razones es que la ejecución de estos sistemas pueden requerir una gran carga computacional, ya que necesitan información de la escena e información previa del objeto. Por lo anterior, es necesario reducir en lo posible el costo computacional para que el sistema funcione a una velocidad de ejecución alta. En

este trabajo se utilizó un método de seguimiento de objetos basado en patrones y modelos de apariencia, ya que es muy utilizado debido a su sencillez y bajo coste computacional. Dicho método se basa en la utilización de patrones o características del objeto para prever el movimiento del objeto a través de cuadros consecutivos [37].

Muchos de los sistemas de seguimiento de objetos mediante múltiples cámaras, utilizan cámaras cuyos campos de visión están compartidos entre sí. Esto presenta ventajas e inconvenientes. Por un lado, al contar con varias vistas de los objetos es posible extraer información sobre la estructura tridimensional de la escena. Por otro, se genera cierta ambigüedad a la hora de seguir los objetos, puesto que un mismo objeto puede tener aspecto diferente bajo distintos puntos de vista. Es por tanto necesario identificar las múltiples proyecciones de los objetos como el mismo elemento 3D, y etiquetarlas de manera consistente en todas las cámaras.

En este trabajo los campos de visión de las cámaras están solapados, permitiendo obtener la información sobre la estructura tridimensional de la escena. Por ello, se propone utilizar dicha información para seleccionar en cada iteración la cámara que tiene mejor vista del objeto, debido a que típicamente las aplicaciones de vídeo vigilancia presentan las imágenes de todas las cámaras a un observador para su análisis. No obstante, la capacidad de una persona para concentrarse en varios vídeos de forma simultánea es muy limitada.

En este capítulo, se presenta una metodología de solución para el problema del seguimiento de objetos en una red de cámaras, describiendo cada uno de los módulos utilizados y cómo éstos interactúan entre sí.

3.1. Descripción del Sistema Propuesto

Los métodos para el seguimiento continuo de objetos en múltiples cámaras requieren de varios módulos de procesamiento que pueden abarcar de forma general: El seguimiento visual de objetos en cada cámara y la fusión de la información disponible por todas las cámaras. Adicionalmente, en este trabajo se propone un método que permite seleccionar en cada iteración la cámara con mejor vista del objeto. La implementación de estos módulos implica el estudio de áreas de matemática y geometría, además del conocimiento en visión por computadora.

Para llevar a cabo el seguimiento continuo de objetos en una red de cámaras, el sistema se dividió en dos etapas principales: En la primera etapa cada cámara realiza el seguimiento del objeto de interés de manera independiente. Posteriormente, en la segunda etapa, se obtiene información de cada cámara con la finalidad de etiquetar los objetos de manera consistente en las mismas. Además, la información espacial y visual de los objetos es utilizada para seleccionar en cada iteración la cámara con mejor vista del objeto. En la Figura 3.2 se puede observar la metodología propuesta y la conexión entre las etapas.

Considerando lo anterior, en este proyecto de tesis se desarrolló un sistema

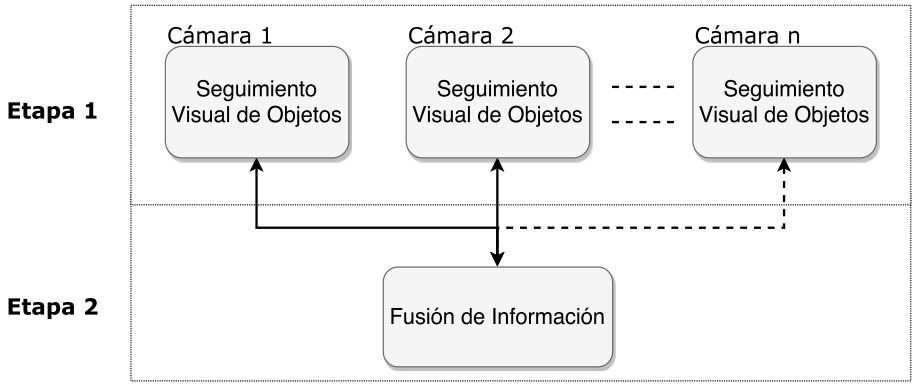

Figura 3.2: *Sistema propuesto para el seguimiento continuo de objetos en una red de cámaras.*

para el seguimiento continuo de objetos en una red de cámaras modeladas, con campos de visión compartida. El sistema inicialmente requiere seleccionar una zona de color homogéneo del objeto de interés en una cámara. Posteriormente, el sistema identifica dicho objeto en las otras cámaras utilizando la información de color del mismo, a través de las relaciones geométricas entre las mismas. Una vez identificado el objeto, se realiza el seguimiento continuo de éste en cada cámara de manera independiente. Además, el sistema selecciona y despliega continuamente la imagen de la cámara con mejor vista del objeto, utilizando su información espacial y el área que ocupa en la imagen.

De manera general, el sistema propuesto recibe como entrada en una iteración, una imagen de cada cámara y al pasar por sus respectivos bloques se obtiene como salida una imagen, donde se muestra en mayor tamaño la imagen de la cámara con mejor vista del objeto y en menor tamaño las imágenes de todas las cámaras. En la Figura 3.3 se observa un ejemplo de la salida del sistema.

Figura 3.3: *Ejemplo de la imagen de salida del sistema propuesto. Aquí, la imagen de mayor tamaño representa la cámara con mejor vista del objeto y las imágenes de menor tamaño representan todas las cámaras del sistema.*

En las secciones siguientes, se describen a manera de detalle las tareas que realizan cada una de las etapas mencionadas.

3.2. Seguimiento Visual de Objetos

En esta sección se describe un método para realizar el seguimiento visual de los objetos, basada en los trabajos de Montecillo [23][22]. En la Figura 3.4 se muestra el sistema de seguimiento visual, donde se observan los módulos de inicialización del modelo del objeto, localización del modelo del objeto en la imagen actual, actualización del modelo del objeto y predicción de la dinámica del objeto.

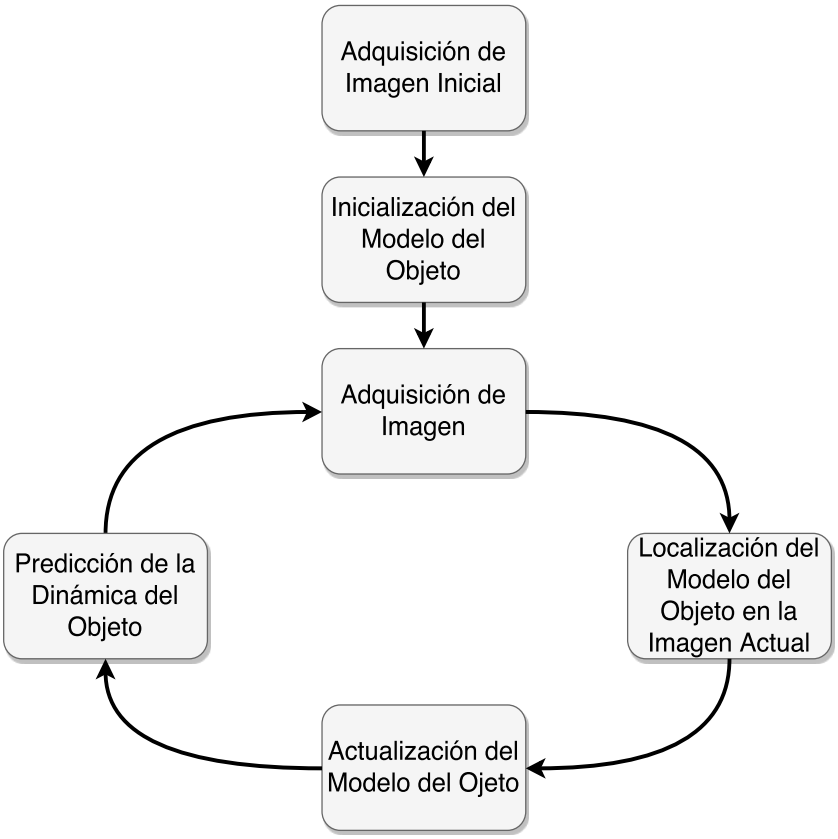

Figura 3.4: *Sistema de seguimiento visual.*

A continuación, se enlistan y se describen los módulos para la implementación del sistema de seguimiento visual.

3.2.1. Problemas asociados a la adquisición de imagen

En los sistemas que involucran cámaras aparecen interferencias propias del sensor que afectan negativamente el desempeño del sistema. Estas interferencias

son defectos inherentes al sensor originadas en su fabricación. Por lo que, resulta importante conocer estos defectos con la finalidad de atenuarlos en lo posible.

Como se mencionó en el capítulo anterior, las cámaras pueden ser modeladas matemáticamente. Este modelo permite estimar los parámetros asociados a su configuración interna, incluyendo algunos defectos en su fabricación. Sin embargo, generalmente dicha calibración es calculada a partir del análisis de imágenes obtenidas previamente por el sensor, ocasionando que el error incluido en dichas imágenes afecte el cálculo del modelo.

Otro de los factores que pueden afectar negativamente el desempeño en los sistemas de seguimiento visual es el ruido presente en el escenario. Este ruido puede ser: la presencia de oclusiones parciales y totales, y los cambios de iluminación.

Las oclusiones suponen uno de los principales problemas en estos sistemas, puesto que en muchas ocasiones la parte visible del objeto a reconocer es reducida. Por otra parte, los cambios de iluminación también pueden ocasionar grandes problemas debido a que pueden crear sombras o reflejos en los objetos, y producir interferencias importantes en el sistema.

3.2.2. Modelo del objeto

Es muy importante modelar de manera apropiada el objeto detectado con el fin de realizar un seguimiento fiable, es por eso que cada tipo de modelo se ajusta más a un objetivo que a otro. Por ejemplo, se utilizan distintas tipologías de representación para sistemas de detección de matrículas que para seguimiento de personas o vehículos.

Los diversos modelos que existen para representar objetos se pueden dividir en dos grandes grupos: modelos basados en formas y modelos basados en apariencia (ver Figura 3.5) [37].

En este trabajo se utilizó un modelo basado en apariencia, específicamente un modelo de densidad de probabilidad, basado en los trabajos de Montecillo [23][22]. Observando que dicho modelo intenta atenuar los efectos negativos que ocasionan los cambios de iluminación en el objeto, utilizando espacios de color perceptual con la finalidad de separar la componente de luminancia del color. Además, ofrece un costo computacional relativamente bajo.

Los objetos son modelados utilizando su color en un espacio de color perceptual como HSI o CIELab, seleccionando inicialmente una zona homogénea del objeto (ver Figura 3.6), observando que en dicho modelo cada componente de color de la zona seleccionada es representado por un conjunto difuso de tipo triangular definido mediante su límite inferior, superior y valor modal. Posteriormente, los conjuntos difusos son modificados dinámicamente con la finalidad de robustecer el sistema ante los cambios de iluminación.

El tipo de conjunto difuso utilizado en este trabajo es el Gaussiano (definido por el valor medio y la desviación estándar de cada componente de color de la zona seleccionada del objeto), debido a que se observó que los valores numéricos

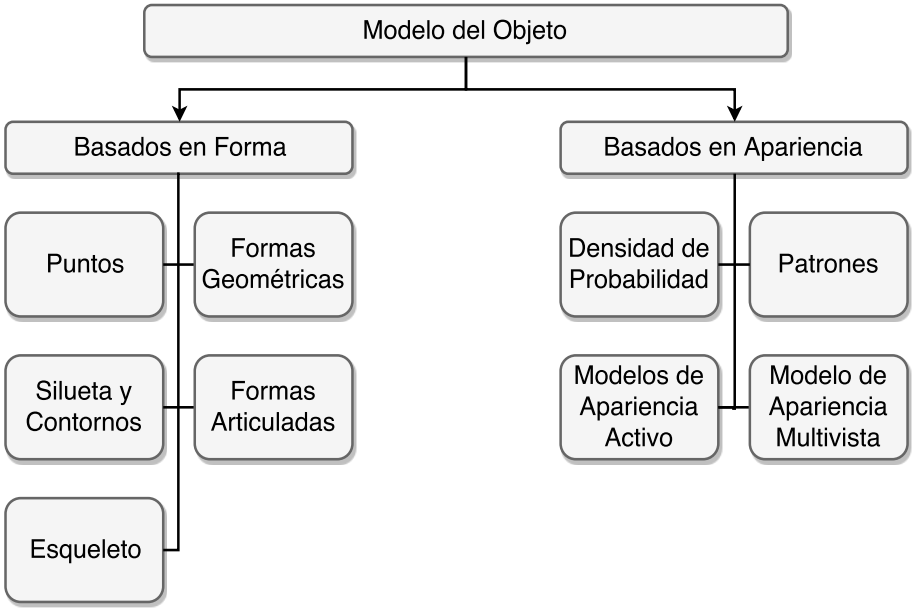

Figura 3.5: *Clasificación de los métodos de modelado de objetos [37].*

Figura 3.6: *Muestra del color objetivo (recuadro en la esfera).*

de las componente de color del objeto en cuadros consecutivos pueden variar considerablemente. Esto ocasiona que, dependiendo de la iteración, la selección inicial de la zona de color del objeto, genere diferentes valores en los conjuntos difusos. Por lo que, la función Gaussiana fue seleccionada al ser simétrica y representar de buena forma la distribución de valores aleatorios alrededor de una media, como lo es el color de una zona aparentemente homogénea.

3.2.3. Estrategias de búsqueda e identificación del modelo en la imagen

Esta etapa busca determinar cuál región dentro de la zona de búsqueda pertenece al objeto de interés. Esta región se obtiene evaluando los píxeles en los conjuntos difusos del modelo del objeto.

La evaluación de los píxeles en el modelo del objeto consiste en decidir si un

píxel p cualquiera dentro de la región de búsqueda pertenece o no al color objetivo. Esta decisión se logra combinando los conjuntos difusos de las componentes de color usando operadores difusos, como se observa en [23][22]. Por lo que, si el grado de pertenencia del píxel p es igual o mayor a un umbral establecido, éste es considerado parte del objeto de interés. El umbral seleccionado en este trabajo fue de 0.7.

Una vez evaluados los píxeles dentro de la zona de búsqueda, se obtienen regiones de color que pueden presentar ruido o huecos entre regiones. Por ello, en este trabajo se utilizaron operaciones morfológicas de erosión y dilatación. Inicialmente, se aplica el proceso de erosión con un elemento estructurante rectangular del tamaño de 4 píxeles, eliminando regiones menores a dicho tamaño que el sistema considera como parte del objeto pero que no pertenecen al mismo. Luego, la dilatación es aplicada con un elemento estructurante rectangular del tamaño de 10 píxeles, con la finalidad de expandir las regiones que contengan al menos un píxel en dicho tamaño, uniendo regiones próximas entre sí.

Finalmente, se selecciona la región resultante con el área más grande, con la finalidad de evitar zonas ajenas al objeto. La región seleccionada es delimitada por un recuadro que indicará la ubicación actual del objeto y la zona utilizada para la predicción de la dinámica del objeto.

3.2.4. Estrategias de actualización del modelo

Muchos de los escenarios utilizados para el seguimiento de objetos presentan cambios considerables de iluminación. Estos cambios ocasionan que los valores numéricos de los componentes de color del objeto varíen en cuadros consecutivos. El modelo del objeto utilizado permite manejar cambios menores de iluminación. Sin embargo, cuando la iluminación cambia drásticamente de un cuadro al siguiente, el color también lo hace, causando fallas en la descripción del objeto usando este modelo.

La evaluación de los píxeles en los conjuntos difusos permiten conocer el grado de similaridad o pertenencia que tienen con el color objetivo. Esta similitud es afectada por los cambios de iluminación, ya que los nuevos valores de color pueden estar fuera del rango permitido por los conjuntos difusos. Por lo anterior, se hace necesario cambiar los parámetros de estos conjuntos para reducir el efecto negativo causado por la variación de la iluminación. En este trabajo se utilizó la estrategia de actualización del modelo encontrada en [23][22], donde los parámetros de cada conjunto difuso son modificados en cada cuadro para adaptarse a estas variaciones.

La adaptación de los conjuntos difusos a los cambios de iluminación consiste esencialmente en desplazar horizontalmente estos conjuntos según un valor X_c' calculado, como se observa en la Ecuación 3.1.

$$X_c' = X_c + \gamma \Delta R \tag{3.1}$$

donde, X_c representa el valor actual de la media de la función gaussiana. ΔR

corresponde a la diferencia entre la media X_c de la función gaussiana y el promedio de píxeles que superan un umbral de 0.7. El valor de γ se eligió diferente para cada componente de color, debido a que la componente más afectada por los cambios de iluminación es I y L en los modelos de color HSI y CIELab, respectivamente. Por lo tanto, para la componente de color I y L se utilizó un valor de γ de 0.4 y, para las otras componentes 0.1.

De manera general, si el valor de ΔR es positivo el desplazamiento es hacia la derecha de X_c y si es negativo hacia la izquierda. La Figura 3.7 muestra cómo se desplaza alguna función de membresía gaussiana con respecto al parámetro X_c.

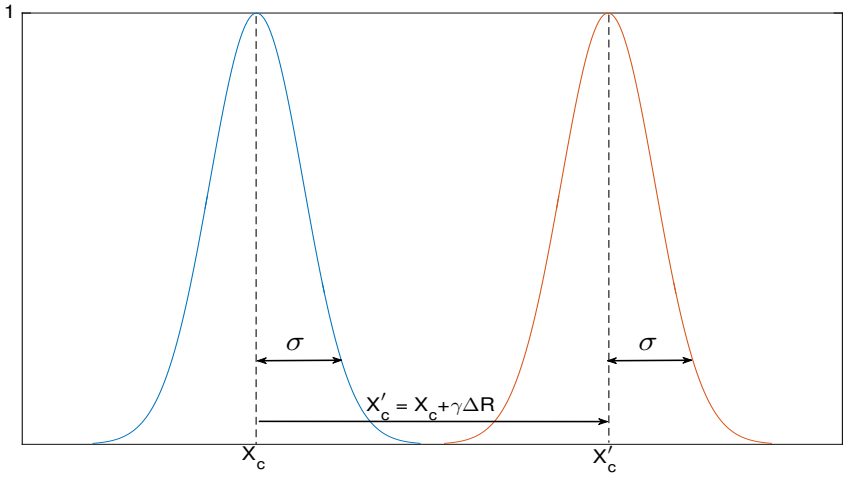

Figura 3.7: *Desplazamiento de una función de membresía.*

3.2.5. Estrategia de predicción de la dinámica del objeto

Una vez modelados los objetos, es necesario estimar continuamente sus posiciones utilizando técnicas de predicción. Éstas técnicas deben atenuar el ruido presente en la estimación de la posición de los objetos con la finalidad de encontrar las zonas donde éstos están incluidos en cada iteración. Sin embargo, estos sistemas pueden ser afectados por muchos factores que reducen el desempeño de los mismos, estos pueden ser: cambios de iluminación, oclusión, zona de búsqueda muy pequeña, velocidad del objeto alta o que el objeto esté ubicado fuera de la zona de búsqueda.

La estimación de la posición de los objetos puede ser calculada principalmente de dos maneras, con predicción o sin predicción. Los sistemas de seguimiento de objetos que no utilizan predicción, generalmente utilizan la posición actual del objeto como centro de la zona donde posiblemente se encuentre el objeto en una posición posterior. Esto tiene el inconveniente de que si el objeto no se encuentra dentro de la zona de búsqueda, el sistema no puede localizarlo y, por consiguiente,

pierde su ubicación. Por esto, una de las principales problemáticas en los sistemas sin predicción es que la zona de búsqueda no contenga al objeto de interés. Esta problemática se ilustra en la Figura 3.8.

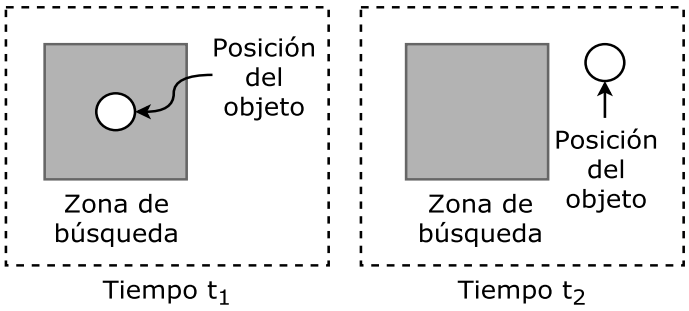

Figura 3.8: *Zona de búsqueda sin predicción.*

El problema mencionado puede atenuarse utilizando técnicas de predicción que permitan estimar la ubicación de los objetos de mejor manera, como se observa en la Figura 3.9. El filtro de Kalman es ampliamente utilizado para resolver esta problemática, ya que puede realizar predicciones y tener un buen manejo de las mediciones ruidosas. Este filtro produce una estimación óptima estadísticamente, tomando en cuenta las mediciones y la información a priori que se tenga del sistema.

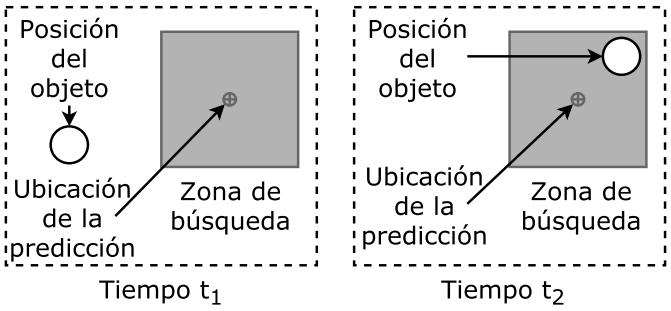

Figura 3.9: *Zona de búsqueda con predicción.*

En este trabajo se utilizó el filtro de Kalman basado en las implementaciones de Montecillo [23][22], con la finalidad de predecir la zona de búsqueda donde posiblemente se encuentra el objeto de interés en cada iteración, a través de la estimación de la posición del centro del objeto y su tamaño visual.

Predicción de posición y tamaño

El filtro de Kalman proporciona un algoritmo recursivo de varianza de error lineal, imparcial y mínimo, para estimar óptimamente el estado desconocido de un sistema dinámico a partir de datos ruidosos tomados en tiempo real (discreto).

Este filtro aborda el problema general de intentar estimar las variables de estado de un proceso controlado en tiempo discreto.

Las ecuaciones utilizadas para implementar el filtro de Kalman se dividen en dos grupos: Ecuaciones de predicción y ecuaciones de actualización. Las ecuaciones de predicción son responsables de proyectar hacia delante (en el tiempo) las estimaciones de los errores del estado actual para obtener las estimaciones a priori del siguiente paso de tiempo. Las ecuaciones de actualización son responsables de la realimentación, es decir, incorporan una nueva medida en la estimación a priori para obtener una estimación a posteriori mejorada [36].

El proceso completo que utiliza el filtro de Kalman para la predicción de los estados en cada iteración se puede evidenciar en la figura 3.10.

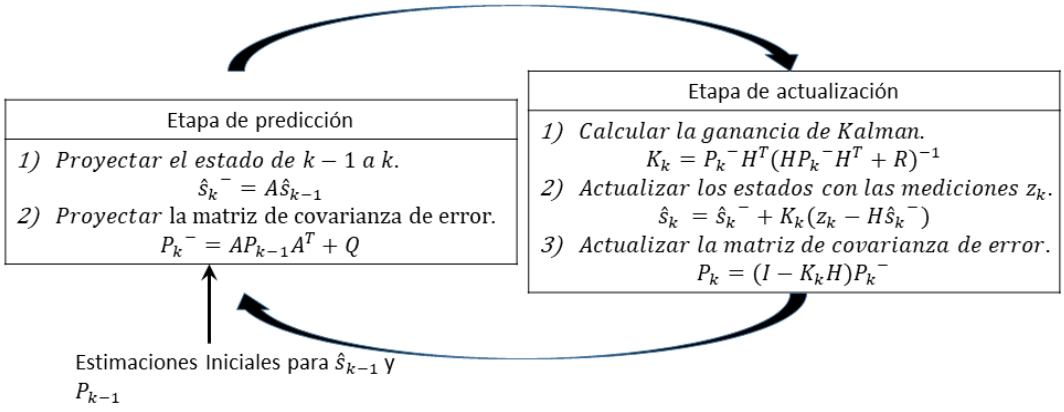

Figura 3.10: *Operación completa del filtro de Kalman.*

Para la implementación del filtro de Kalman se deben conocer las siguientes matrices:

- La matriz A de dimensión $n \times n$, que relaciona el estado de las variables a estimar en la iteración de tiempo anterior $k-1$ con la iteración actual k.
- La matriz H de dimensión $m \times n$, que representa las observaciones o mediciones.
- La matriz de covarianza Q de dimensión $n \times n$, que representa el ruido que perturba los estados.
- La matriz de covarianza R de dimensión $m \times n$, que representa el ruido que perturba las mediciones.
- La matriz de covarianza P de dimensión $n \times n$, que representa el error de estimación de los estados.

Inicialmente, se calcula la matriz A suponiendo que el desplazamiento del objeto es de velocidad constante. Los estados son la posición (x, y) del centroide del objeto, el tamaño (n, m) donde n y m representan el ancho y alto del objeto respectivamente y las velocidades (u, v) siendo u, la velocidad del objeto en x y v, la velocidad en y. En la Ecuación 3.2 se observan las definiciones de las variables de estado.

$$\begin{aligned} x_k &= x_{k-1} + u_{k-1}\Delta t \\ y_k &= y_{k-1} + v_{k-1}\Delta t \\ u_k &= u_{k-1} \\ v_k &= v_{k-1} \\ n_k &= n_{k-1} \\ m_k &= m_{k-1} \end{aligned} \tag{3.2}$$

Donde, Δt es el tiempo en que se adquiere una imagen y k la iteración actual del proceso. El sistema de ecuaciones de las variables de estado puede ser escrita como se muestra en la Ecuación 3.3.

$$s_k = As_{k-1} \tag{3.3}$$

Donde, $s_{k-1} = [x\ y\ u\ v\ n\ m]^T$ y A se muestra a continuación.

$$A = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{3.4}$$

Las mediciones registradas en el sistema de estados son la posición y el tamaño del objeto. Por lo que, la matriz H se ilustra en la ecuación 3.5.

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{3.5}$$

En la práctica, las matrices de covarianza de error Q y R pueden cambiar con cada iteración o medida, pero en este trabajo se asumen constantes. Estas matrices Q y R se calcularon utilizando como base los resultados prácticos y lo reportado en la literatura. Para la matriz Q se utilizó 1.0 de varianza en la posición, tamaño y velocidad, mostrada a continuación.

$$Q = \begin{pmatrix} 1.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 \end{pmatrix} \tag{3.6}$$

Para la matriz R, asociada al error que tienen las mediciones, se estima que la varianza es de un pixel, por lo tanto

$$R = \begin{pmatrix} 1.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 \end{pmatrix} \tag{3.7}$$

Finalmente, la matriz P que depende directamente de Q y R fue calculada utilizando los resultados prácticos y asumiendo que la velocidad puede presentar mayor variación en sus valores. Para la posición y tamaño se utilizó 9.0 de varianza y para la velocidad 25.0 de varianza. La matriz P se observa a continuación.

$$P = \begin{pmatrix} 9.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 9.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 25.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 25.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 9.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 9.0 \end{pmatrix} \tag{3.8}$$

Las ecuaciones mostradas en esta sección pueden ser revisadas a un mayor nivel de detalle en [17][10].

3.3. Fusión de Información

En los sistemas de seguimiento mediante múltiples cámaras es necesario identificar las múltiples proyecciones de los objetos como el mismo elemento, y etiquetarlas de manera consistente en todas las cámaras. Por lo que, en este trabajo se propone un método de solución a este problema. Además, se propone un método que permite utilizar la información disponible por todas las cámaras para seleccionar continuamente la cámara con mejor vista del objeto.

Para implementar los métodos discutidos en esta sección, es necesario inicialmente modelar las cámaras obteniendo la matriz intrínseca y extrínseca de las mismas, sobre un marco de coordenadas global. Para ello, en este trabajo se utilizó el *toolbox* de calibración de cámaras desarrollado por Jean-Yves Bouguet [3] en la herramienta de software matemático MATLAB.

En la Figura 3.11 se observan las etapas utilizadas para describir dichos métodos, siendo estas, la agregación de la información disponible y la selección de la información.

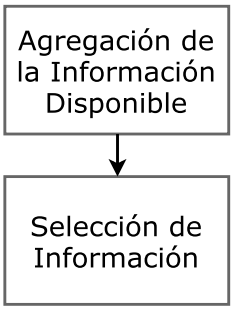

Figura 3.11: *Sistema de fusión de información.*

A continuación, se desarrollan las etapas para la implementación del sistema de fusión de información.

3.3.1. Agregación de la información disponible

Una vez resuelto el problema del seguimiento visual de los objetos en una cámara, se procede a utilizar la información disponible en el sistema para ubicar y modelar dichos objetos en las otras cámaras. Sin embargo, esto resulta una tarea difícil de realizar debido a que los objetos pueden presentar variaciones en sus características cuando son observados desde diferentes vistas. En la Figura 3.12 se observan los módulos utilizados para ello.

Figura 3.12: *Sistema para la agregación de la información disponible.*

En esta sección, la información disponible del objeto en una cámara, es utilizada para encontrar inicialmente la zona donde posiblemente se encuentre dicho objeto en otra cámara, y posteriormente, modelarlo y seguirlo. Este proceso se muestra a continuación.

Información del objeto

Inicialmente, para poder estimar la ubicación y el modelo del objeto en las cámaras donde éste no ha sido identificado, se requiere conocer información a priori de dicho objeto. Esta información puede ser extraída de las cámaras donde el objeto fue modelado previamente. En este trabajo, la información extraída de estas cámaras es el modelo del objeto y el área que ocupa en la escena.

La información del modelo extraído de las cámaras donde el objeto fue identificado, incluye la media X_c y la desviación estándar σ del conjunto difuso de cada componente de color, discutido anteriormente.

El área que ocupa el objeto en la escena, se logra construyendo una región de píxeles que pertenecen al objeto dentro de una zona de búsqueda, como se mostró en la sección 3.2.3. En la Figura 3.13 se observa un ejemplo de esta región.

(a) *Objeto de interés.* **(b)** *Región de píxeles que pertenecen al objeto.*

Figura 3.13: *Región utilizada por la geometría epipolar.*

Determinación de la zona de búsqueda del objeto

Una vez obtenida la información a priori de los objetos previamente identificados, se puede construir las zonas de búsqueda que incluyen dichos objetos. Estas zonas deben delimitar en lo posible solo los objetos de interés, en base a su tamaño visual.

Para ubicar y modelar los objetos, se estima inicialmente la zona donde posiblemente se encuentre, con la finalidad de evitar regiones que tengan características similares a dicho objeto. Sin embargo, esta zona puede variar de una cámara a otra debido a la ubicación y orientación de las mismas. Estas zonas de búsqueda se obtuvieron utilizando las restricciones de la geometría epipolar y características de movimiento (flujo óptico).

La geometría epipolar [15] permite proyectar un punto de una cámara a una recta en otra cámara, a través de la matriz fundamental. Esta matriz es calculada utilizando la información obtenida del *toolbox* de calibración de cámaras [3] al corresponder entre cada par de cámaras, los puntos de interés de un patrón de calibración.

Las proyecciones de la geometría epipolar hace posible que, a partir de algunos puntos seleccionados del objeto en una cámara, se construya una región en otra cámara que incluya dicho objeto. Sin embargo, esta región puede abarcar gran parte de la escena, ocasionando que el sistema pueda confundir alguna zona de la escena con el objeto de interés, por lo que se hace necesario delimitar dicha región. El flujo óptico puede ser usado para delimitar esta región, ya que permite estimar zonas en la escena que presentan movimiento. Por lo tanto, la intersección de la región generada por la geometría epipolar y las zonas de movimiento en la escena, originan zonas de búsqueda de menor tamaño que pueden contener al objeto de interés.

La técnica de flujo óptico utilizado en este trabajo es el de Farnebäck [14], por su fácil implementación y buen compromiso entre velocidad y precisión [35]. Esta técnica calcula el movimiento relativo de cada píxel en la escena por lo que puede aumentar considerablemente el costo computacional del sistema por la gran cantidad de cálculos que realiza en cada iteración. Sin embargo, esta técnica puede ser prescindida si el objeto es identificado en dos o mas cámaras. Esto es posible debido a que la intersección de las regiones generadas por la geometría epipolar desde dos o más cámaras pueden proveer una zona de búsqueda que delimite al objeto de interés.

En este trabajo se utilizaron cuatro puntos del objeto que son proyectados a otras cámaras, siendo estos puntos, el punto centro-superior, centro-inferior, centro-izquierdo y centro-derecho de la zona visual que ocupa el objeto en la escena. Esta selección se hizo debido a que estos puntos representan los límites visuales del objeto. En la Figura 3.14, se observa un ejemplo de la selección de dichos puntos sobre un balón.

Figura 3.14: *Selección de los puntos utilizados por la geometría epipolar.*

Posteriormente, los puntos seleccionados en la Figura 3.14 son proyectados como líneas en las otras cámaras (ver Figura 3.15a). La región entre estas líneas genera la zona de búsqueda requerida (ver Figura 3.15b).

Como se observa en la Figura 3.15b, la zona de búsqueda abarca un área importante de la escena, pudiendo incluir regiones que tengan características de color similares al del objeto de interés. Por esto, la zona de búsqueda generada por el flujo óptico es utilizada para evitar en lo posible estas regiones.

Finalmente, las regiones formadas a través del flujo óptico por el agrupamiento de los píxeles que tienen una magnitud de desplazamiento mayor a tres píxeles, son consideradas zonas de búsqueda. Por lo que, la intersección de la zona de búsqueda generada por la geometría epipolar y el flujo óptico, originan la zona de búsqueda final que delimita el objeto de interés (ver Figura 3.16).

(a) *Lineas epipolares proyectadas de los puntos seleccionados en la Figura 3.14.*

(b) *Zona de búsqueda umbralizada.*

Figura 3.15: *Zona de búsqueda generada por la geometría epipolar.*

Identificación del objeto

Una vez resuelto el problema de encontrar la zona de búsqueda en las cámaras donde el objeto no ha sido identificado, el último problema a resolver, para poder identificar estos objetos, es el de obtener el modelo del objeto y la zona visual que ocupa en la escena. Este problema consiste básicamente en encontrar dentro de la zona de búsqueda los píxeles que pertenecen al objeto. Para resolver esto, se utilizó la variante del algoritmo de segmentación por regiones $Watershed$ desarrollada por F. Meyer [21], con el objetivo de dividir la zona de búsqueda en regiones de color uniforme y encontrar los píxeles que pertenecen al objeto.

Este algoritmo permite segmentar la imagen a partir de los marcadores seleccionados. En este trabajo, los marcadores son elegidos automáticamente utilizando el modelo del objeto en las cámaras donde éste fue identificado. Este modelo consta de un conjunto difuso por cada componente de color. Sin embargo,

(a) *Intersección de las zonas de búsqueda.* **(b)** *Zona de búsqueda final (perímetro rojo).*

Figura 3.16: *Zona de búsqueda final.*

las componentes I y L de los modelos de color HSI y CIELab respectivamente, no son utilizadas en esta etapa debido a que representan la iluminación percibida por el objeto, pudiendo tener valores muy diferentes en cada una de las cámaras. Por lo tanto, los marcadores son seleccionados dentro de la zona de búsqueda evaluando en el modelo del objeto los píxeles que tienen un grado de pertenencia mayor a 0.6.

Una vez segmentada la imagen, se debe verificar que efectivamente la región obtenida pertenezca al objeto. Esta verificación se realiza comparando los histogramas de las componentes de color de la región obtenida (H_1), con los histogramas de las componentes de color utilizadas para la segmentación (H_2). Dicha comparación se realizó utilizando la distancia de Bhattacharyya [2] (ver Ecuación 3.9) debido a que incorpora en su cálculo la desviación estándar de las distribuciones de los píxeles.

$$\begin{aligned} d(H_1, H_2) =& \sqrt{1 - \frac{1}{\sqrt{\bar{H}_1 \bar{H}_2 N^2}} \sum_I \sqrt{H_1(I) H_2(I)}}, \\ \bar{H}_k =& \frac{1}{N} \sum_J H_k(J) \end{aligned} \tag{3.9}$$

Donde, N es el número total de celdas de los histogramas. Esta distancia toma valores en el rango de 0 a 1, siendo 0 el valor de máxima similitud y 1 el valor de máxima discordancia. En este trabajo, la región segmentada es considerada parte del objeto si el valor de la distancia es menor a 0.4, en caso contrario la región es descartada y el proceso se repite en la siguiente iteración. Una vez verificada la región, el objeto es identificado, obteniendo su modelo y la zona que ocupa en la escena (ver Sección 3.2.2).

3.3.2. Selección de información

Continuando con las etapas del sistema de fusión de información mostrado en la Figura 3.11, la etapa de selección de información permite considerar la información disponible por el sistema con la finalidad de realizar la selección de la cámara con mejor vista del objeto. En la metodología propuesta son utilizadas características visuales y de distancia espacial como información principal para dicha selección. Además, se propone la combinación de estas características para intentar predecir la selección natural de un usuario.

La metodología propuesta consiste esencialmente en utilizar el área visual del objeto en las cámaras donde está identificado. En este trabajo, una de las formas para seleccionar la cámara con mejor vista del objeto, es encontrar la cámara que tenga el mayor área visual de dicho objeto. Sin embargo, esto no indica necesariamente que dicha cámara ofrece la mejor vista del objeto, debido a que las oclusiones parciales pueden reducir el área visual del objeto, ocasionando fallas en la selección. Por ello, en este trabajo, la distancia espacial del objeto a cada cámara es considerada para reducir en los posible estas fallas.

La distancia espacial del objeto de interés a cada cámara, es calculada utilizando la triangulación 3D. Esta triangulación consiste en encontrar las coordenadas espaciales de los píxeles de interés sobre un marco de coordenadas global, utilizando la correspondencia de éstos en dos cámaras, y los parámetros intrínsecos y extrínsecos de dichas cámaras. Esto puede ser visto a un mayor nivel de detalle en [15].

La triangulación permite obtener la ubicación 3D de un punto en el escenario, visto desde dos cámaras. Donde, dicho punto corresponde al centroide del área visual del objeto. A continuación, se describen los pasos para calcular la distancia espacial del objeto a cada cámara. Estos pasos toman en cuenta solo las cámaras donde el objeto está identificado.

1.- Se selecciona al azar una cámara como referencia para hacer la triangulación 3D.

2.- Posteriormente, el centroide del objeto en la cámara seleccionada es proyectado como líneas en las demás cámaras, utilizando la geometría epipolar, mencionada anteriormente.

3.- Seguidamente, las líneas proyectadas son consideradas para encontrar la correspondencia del centroide de la cámara de referencia en las otras cámaras. Esto se debe a que no necesariamente los centroides corresponden entre sí en todas las cámaras.

    Inicialmente, se calcula la recta que es perpendicular a la línea proyectada y que a su vez pasa por el centroide del objeto en la cámara estudiada, utilizando la Ecuación punto-pendiente 3.10.

$$y - y_1 = -\frac{1}{m}(x - x_1) \tag{3.10}$$

Donde, m es la pendiente de la línea proyectada, y (x_1, y_1) es la coordenada del centroide del objeto en la cámara estudiada (Punto A). Una vez obtenida esta recta, se procede a calcular el punto de intersección de ésta con la línea proyectada (Punto P), igualando sus ecuaciones. Este punto es el utilizado para hacer la triangulación 3D con el centroide del objeto en la cámara de referencia. En la Figura 3.17 se observa gráficamente este paso.

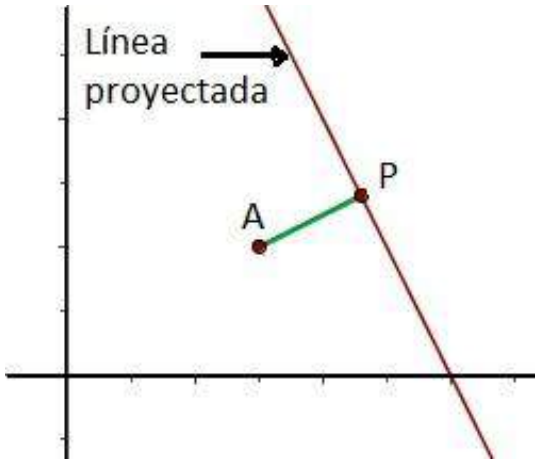

Figura 3.17: *Proceso de correspondencia de los centroides en dos cámaras.*

4.- Una vez obtenidos los puntos de correspondencia en todas las cámaras, se procede a realizar la triangulación 3D. La triangulación es calculada para cada cámara utilizando como referencia el centroide del objeto en la cámara seleccionada en el primer paso. Por lo tanto, son obtenidos $n - 1$ puntos 3D del escenario, donde n es el número de cámaras.

5.- Los puntos 3D obtenidos, son verificados evaluando el error en el cálculo de los mismos. Esto se hace proyectando cada punto 3D en la imagen de la cámara donde fue obtenido, a través de la Ecuación 2.3. Posteriormente, se calcula la distancia euclidiana entre el punto proyectado en la imagen y el encontrado en el paso tres. Esta distancia indica el error en píxeles en el cálculo del punto 3D. Por lo que, a menor distancia, mejor fue la triangulación. En este trabajo, los puntos 3D que presentan un error menor a 30 píxeles, son los utilizados en pasos posteriores.

6.- Luego de verificar los puntos 3D, se promedian, utilizando las siguientes ecuaciones:

$$X = \frac{\sum_i x_i}{N},\ Y = \frac{\sum_i y_i}{N},\ Z = \frac{\sum_i z_i}{N}.$$

Donde, x_i, y_i y z_i corresponden a las coordenadas del punto 3D en la $i-esima$ cámara, N es el número total de puntos 3D, y X, Y y Z es el punto 3D que representa la ubicación del objeto en el marco de coordenadas global.

7.- Una vez obtenida la ubicación del objeto en el marco de coordenadas global, basta encontrar la ubicación de las cámaras en dicho marco de coordenadas, para así, obtener la distancia del objeto a cada cámara.

La ubicación de las cámaras en el marco de coordenadas global, es encontrada utilizando la matriz de calibración extrínseca de cada cámara (ver Ecuación 2.2). En la Ecuación 3.11, se observa una forma de representar esta matriz, permitiendo proyectar un punto 3D en el marco de coordenadas global (X, Y, Z), a un punto 3D en el marco de coordenadas del centro de la cámara (x, y, z), como se discutió en el capítulo anterior.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} * R + t \tag{3.11}$$

Sin embargo, como la ubicación de la cámara debe estar referenciada al marco de coordenadas global, la ecuación anterior es despejada como se muestra en la Ecuación 3.12. En esta Ecuación se observa que basta con calcular el vector de traslación $-t * R^{-1}$ para encontrar la ubicación de la cámara en el marco de coordenadas global.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} * R^{-1} - t * R^{-1} \tag{3.12}$$

8.- Finalmente, teniendo la ubicación del objeto y las cámaras, en el marco de coordenadas global, se calcula la distancia (D_i) del objeto a la cámara i, utilizando la ecuación 3.13.

$$D_i = \sqrt{(C_x^i - O_x)^2 + (C_y^i - O_y)^2 + (C_z^i - O_z)^2} \tag{3.13}$$

Donde, $C = (-t * R^{-1})^T$ y O (punto obtenido en el paso 6) representan la ubicación 3D de la cámara y del objeto respectivamente, sobre el marco de coordenadas global.

La distancia del objeto a cada cámara, se utiliza para identificar la cámara que se encuentra más cerca del mismo en cada iteración. Sin embargo, al igual que el problema del área visual, no necesariamente la cámara más cerca del objeto ofrece la mejor vista del mismo, debido a que la orientación del objeto puede ocasionar que el mismo no se observe de la mejor manera, ocasionando fallas en la selección de la cámara con mejor vista. Por lo que, en este trabajo se propone la combinación de ambas características de información para seleccionar la cámara con mejor vista del objeto.

Selección de la cámara con mejor vista del objeto

Como se mencionó anteriormente, la utilización de la información espacial y visual del objeto, presentan desventajas que pueden afectar negativamente el desempeño en el proceso de selección de la cámara con mejor vista del objeto. Por lo que, la combinación de éstas puede mejorar dicha selección.

En este trabajo, la selección de la cámara con mejor vista del objeto fue realizada utilizando la Ecuación 3.14. Donde, la información de distancia espacial (D_i) del objeto de interés a la cámara i, y el área visual (A) del mismo en dicha cámara, son normalizados en el rango de 0 a 1 utilizando la información de las cámaras donde el objeto está identificado. Además, con la finalidad de obtener un valor (d_i) que represente la métrica de selección en cada cámara, es utilizada la constante de proporción α, obtenida de manera empírica según la disposición de las cámaras en el escenario.

Note que en la Ecuación 3.14, la información de distancia espacial es representada como $1 - D_i$, ya que se considera que mientras menor sea la distancia entre el objeto y la cámara i, mejor vista del objeto tiene dicha cámara.

$$d_i = (1 - \alpha) * (1 - D_i) + \alpha * A_i \tag{3.14}$$

Finalmente, la selección de la cámara con mejor vista del objeto se realiza, eligiendo la cámara que tenga mayor métrica de selección d.

3.4. Conclusiones del Capítulo

En este capítulo, se propuso una metodología de solución al problema del seguimiento visual de objetos en una red de cámaras. Además, se propuso un método para seleccionar continuamente la cámara con mejor vista del objeto.

En la metodología propuesta se llevó a cabo una descripción detallada de cada uno de sus módulos. Indicando las funciones que realizan cada uno de éstos, así como también sus entradas y salidas mediante las cuales se comunican entre sí.

Capítulo 4

Pruebas y Resultados

Los sistemas de seguimiento de objetos en una red de cámaras son evaluados para determinar la efectividad de su funcionamiento. La precisión, la frecuencia de operación y la robustez, son algunos de los parámetros que se pueden medir para evaluar estos sistemas.

En este capítulo se presentan los resultados obtenidos a partir del método desarrollado en el capitulo anterior. Se realizaron experimentos usando variaciones de los parámetros y distintos espacios de color. Se presentan los mejores resultados y los correspondientes parámetros con los que se obtuvieron esos resultados.

El sistema fue implementado utilizando el lenguaje de programación C++, así como también la biblioteca *OpenCV*, la cual, proporciona un marco de trabajo para el procesamiento de imágenes digitales. Las pruebas se realizaron en una computadora con un procesador Intel Core i5 a 2.5GHz, con 4GB de RAM. Dicho sistema se sometió a diferentes pruebas para determinar su alcance. Las pruebas realizadas consideran los siguientes aspectos:

- Selección de diferentes modelos de color para realizar la representación del objeto.
- Selección de diferentes objetos, tomando en cuenta que la representación del objeto está orientada a objetos con ciertas características.
- Oclusión parcial y total del objeto en las cámaras del sistema
- Cambios aleatorios de iluminación en la escena.
- Estimación de la cámara con mejor vista del objeto. Tomando en cuenta la apreciación de un conjunto de sujetos de prueba.

4.1. Protocolo de Pruebas

Las pruebas realizadas para medir el desempeño del sistema desarrollado se realizaron sobre distintos vídeos. Estos vídeos muestran diferentes condiciones en la escena que puede afectar el funcionamiento del seguimiento visual en múltiples cámaras, como oclusión del objeto, cambios de escala del objeto y cambios de iluminación.

El sistema fue probado utilizando tres vídeos diferentes en entornos de interior y exterior, con la finalidad de evaluar con cada vídeo, uno de los tres módulos principales del sistema, siendo éstos, el seguimiento visual de objetos, la agregación de la información disponible y la selección de la información. Dichos vídeos presentan perturbaciones sobre el objeto de interés, que afectan negativamente el desempeño del módulo en estudio.

Las pruebas realizadas en este capítulo utilizan la métrica F para evaluar cuantitativamente la precisión de las mismas. Esta métrica nos permite saber qué tan certera es la salida definida por el sistema comparada con la verdad de referencia. Además, se calculó el error cuadrático medio de los parámetros obtenidos por el sistema utilizando dicha verdad de referencia para analizar su desempeño.

La verdad de referencia en la mayoría de las pruebas se obtuvo calculando manualmente la región visual que ocupa el objetivo en cada cámara, para todos los cuadros. Adicionalmente, la verdad de referencia de una prueba fue tomada de una base de datos, con el objetivo de realizar su comparación con el estado del arte. Esto se debe a que muchas de las bases de datos usadas en el estado del arte para estos sistemas, utilizan objetos complejos que pueden ocasionar fallas.

Además, en cada prueba se calcula la frecuencia de operación del sistema. Esta frecuencia refleja el tiempo de procesamiento necesario para la correcta ejecución del mismo; esto abarca desde la adquisición del cuadro de cada cámara, hasta el despliegue de la imagen de salida. Sin embargo, este tiempo de procesamiento no es el mismo en cada cuadro debido a diferentes factores, como por ejemplo, el tamaño del área visual del objeto, ya que mientras mas grande sea, mas píxeles deben ser analizados, aumentando el costo computacional. Entonces, la frecuencia de procesamiento calculada es el promedio del tiempo en el que el sistema realiza una iteración.

Las pruebas se realizaron usando los espacios de color HSI y CIELab para comparar su rendimiento y desempeño en el sistema desarrollado.

4.2. Métricas de Evaluación

Una de las métricas utilizadas para evaluar el desempeño del sistema desarrollado, es la métrica conocida como PR (Precision-Recall). Esta métrica realiza una comparación de los parámetros de la verdad de referencia con los

resultados obtenidos por el sistema. Para calcular los parámetros adimensionales P y R se usaron las Ecuaciones 4.1 y 4.2, respectivamente.

$$P = \frac{A}{A+B} \tag{4.1}$$

$$R = \frac{A}{A+C} \tag{4.2}$$

donde, A representa los píxeles relevantes obtenidos, B los píxeles irrelevantes obtenidos y C los píxeles relevantes no obtenidos. En la Figura 4.1, se observa la distribución de A, B y C a partir de los píxeles obtenidos y de la verdad de referencia.

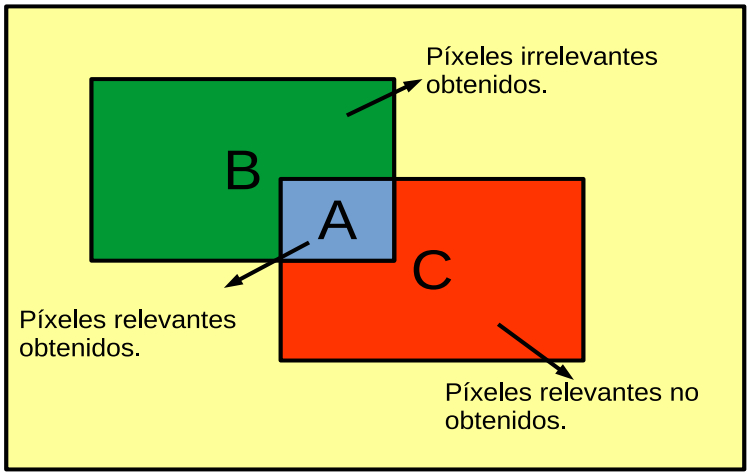

Figura 4.1: *Evaluación de los píxeles encontrados por el sistema comparado con los píxeles de la verdad de referencia.*

La métrica de precisión (Precision) evalúa la fracción de píxeles irrelevantes obtenidos. Esto es, si se obtiene un valor de 1.0, significa que no se encontraron píxeles adicionales comparado con los píxeles de la verdad de referencia. En cambio, si se obtiene un valor de 0.5 indica que la mitad de los píxeles detectados como parte del objeto no pertenecen a los píxeles de la verdad de referencia.

La exhaustividad (Recall) evalúa la fracción de los píxeles relevantes que no fueron detectados por el sistema. Esto es, si el valor es 1.0, significa que ningún píxel de la verdad de referencia fue excluido en la detección.

Otra interpretación de la precisión y la exhaustividad son las siguientes: La precisión es la probabilidad que un píxel obtenido sea un píxel relevante. Por otra parte, La exhaustividad es la probabilidad de que un píxel relevante sea obtenido. Dichas interpretaciones pueden ser combinadas como la media armónica de éstas, comúnmente llamada Métrica F.

De manera general, la métrica F observada en la ecuación 4.3, proporciona un valor en el rango continuo $[0, 1]$ y ofrece un mejor balance de las métricas P y R.

$$F = \frac{2 * P.R}{P + R} \tag{4.3}$$

La métrica F resulta más compleja y se detalla más a fondo en [28]. Sin embargo, lo mencionado anteriormente es suficiente para llevar a cabo la evaluación del sistema desarrollado.

Otra de las métricas utilizadas en este trabajo para evaluar el desempeño del sistema desarrollado, es el cálculo del error cuadrático medio (ECM). Este error permite evaluar los parámetros del objeto obtenidos por el sistema, a través de la comparación de éstos (Ps) con los parámetros de la verdad de referencia (Pvr). En la Ecuación 4.4 se observa el cálculo de dicho error.

$$ECM_x = \frac{1}{n}\sum_{i=1}^{n}(Ps_x - Pvr_x)^2 \tag{4.4}$$

donde, n es el número de cuadros del vídeo estudiado, Ps_x es el parámetro x generado por el sistema y Pvr_x es el parámetro x de la verdad de referencia. ECM_x representa el error cuadrático medio del parámetro x a lo largo del vídeo. Este parámetro puede ser el ancho, alto, centro en x o el centro en y del objeto.

4.3. Pruebas en Vídeos

Los vídeos usados en las pruebas presentan diferentes factores que dificultan el seguimiento visual y la identificación de los objetos en múltiples cámaras. Estas pruebas son evaluadas empleando la métrica F y el error cuadrático medio para calificar el funcionamiento del sistema ante estos factores.

Las pruebas realizadas para evaluar el desempeño del sistema están basadas en la utilización de tres vídeos diferentes. Sin embargo, en cada vídeo son aplicados diferentes niveles de ruido sal y pimienta, con la finalidad de evaluar la robustez de dichas pruebas. De manera general, las pruebas realizadas buscan evaluar las dos etapas principales definidos en el capítulo anterior, éstos son: Seguimiento visual de objetos y la fusión de información.

4.3.1. Seguimiento Visual de Objetos

En el primer vídeo se realiza el seguimiento de una taza de color azul vista desde una cámara, con la finalidad de evaluar el módulo del seguimiento visual de objetos, discutido en el capítulo anterior. Este vídeo y su verdad de referencia fue obtenido de la base de datos $BoBoT$, proporcionada por la Universidad de Bonn en Alemania de nombre Cup [34]. En este vídeo el objeto es manipulado por una mano, donde el movimiento es suave y sin cambios repentinos y rápidos en la posición del mismo. Además, en dicho vídeo se observa una escena de interior, donde los cambios de iluminación son semi-controlados. Sin embargo, se debe

notar que el objeto posee figuras de diferente color y refleja la luz fácilmente, añadiendo ruido al sistema. En el Anexo A.1 se muestra una secuencia exitosa del seguimiento visual de este primer vídeo.

En la Tabla 4.1 se muestran los mejores resultados obtenidos en el análisis del primer vídeo. En dicha tabla se observa que se realizaron cinco pruebas para el espacio de color CIELab y dos pruebas para HSI, debido a que la prueba falla en el espacio de color HSI para ruidos mayores al ruido base. El ruido base corresponde a la adición del ruido sal y pimienta en cada cuadro, donde la probabilidad que un píxel sea blanco o negro es del 1 %. Además, el ruido base x5, x10 y x20, representa que la probabilidad de que un pixel sea blanco o negro es del 5 %, 10 % y 20 %, respectivamente.

Tabla 4.1: *Mejores resultados del análisis del primer vídeo.*

CIELab				
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**	**Frecuencia (Hz)**
Sin ruido	0.543	0.873	0.653	50.2
Ruido base	0.543	0.867	0.658	50.1
Ruido base x5	0.576	0.865	0.681	49.8
Ruido base x10	0.638	0.833	0.717	49.6
Ruido base x20	0.679	0.830	0.742	49.5
HSI				
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**	**Frecuencia (Hz)**
Sin ruido	0.550	0.901	0.675	52.1
Ruido base	0.549	0.899	0.674	52.0

También, en la Tabla 4.1 se puede observar que en estas pruebas el comportamiento del sistema usando el espacio de color HSI resulta similar al espacio de color CIELab. La frecuencia de operación es mayor en el espacio de color HSI, debido a que las operaciones de conversión en este espacio de color son más sencillas. Sin embargo, como se mencionó anteriormente, el espacio de color HSI es más sensible al ruido sal y pimienta. Adicionalmente, en esta tabla se observa que en el espacio de color CIELab a medida que el ruido aumenta, la exhaustividad y la métrica F también lo hacen, mientras que la precisión disminuye. Esto sucede debido a que al seleccionar manualmente la zona de color objetivo, se incluyen más píxeles blancos o negros, aumentando la desviación estándar de los conjuntos difusos estudiados anteriormente. Esto permite obtener una mayor cantidad de píxeles que pertenecen al objeto, cubriendo más área del mismo. Sin embargo, estos píxeles pueden abarcar zonas del escenario, afectando negativamente la precisión.

Para finalizar las pruebas sobre el primer vídeo, se realizó la comparación de algunos parámetros obtenidos por el sistema, con los parámetros de la verdad de referencia, utilizando el error cuadrático medio.

En el Anexo A.2 se muestran cuatro gráficas, en cada gráfica se observan los

valores de la verdad de referencia y los obtenidos por el sistema, en la prueba sin el ruido sal y pimienta. En la Tabla 4.2 se observa el error cuadrático medio de los parámetros observados en dichas gráficas para todas las pruebas, siendo éstos: El centro en x, el centro en y, el ancho y el alto, de la zona que delimita al objeto en unidades de píxeles.

Tabla 4.2: *Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el primer vídeo.*

CIELab				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	67.68	17.66	170.69	108.47
Ruido base	49.95	4.42	168.28	106.04
Ruido base x5	41.10	3.66	132.02	85.53
Ruido base x10	30.27	3.69	114.04	66.07
Ruido base x20	20.91	3.85	90.49	56.78
HSI				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	35.94	2.28	133.02	125.04
Ruido base	35.76	2.15	133.07	126.47

En la Tabla 4.2, se evidencia que al igual de la tabla 4.1 los errores disminuyen, al considerar más píxeles como parte del objeto. Sin embargo, de forma general se observa que los errores obtenidos utilizando el espacio de color HSI son menores.

Por último, en el Anexo A.3 se observa un ejemplo del comportamiento de la adaptación dinámica de los conjuntos difusos en el espacio de color CIELab, en la prueba sin ruido.

4.3.2. Fusión de Información

Continuando con las pruebas realizadas al sistema, se evalúan los métodos propuestos discutidos en la Sección 3.3, asociados en primer lugar al uso de la información disponible por el sistema para etiquetar de manera consistente las múltiples proyecciones de los objetos en todas las cámaras, y en segundo lugar, a la selección de la cámara con mejor vista del objeto.

Agregación de la información disponible

En el segundo vídeo se realiza el seguimiento de una persona a través de su suéter color azul en una escena de exterior vista desde tres cámaras, donde los cambios de iluminación no son controlados y se presentan oclusiones parciales y totales, con la finalidad de evaluar el módulo de identificación de objetos, discutido anteriormente. Este vídeo fue obtenido de la base de datos publicada en [13], y su verdad de referencia fue calculada manualmente para cada cámara. Además,

la base de datos proporciona la calibración del sistema, discutida en el capítulo anterior. En este vídeo, la persona presenta movimientos suaves y sin cambios repentinos y rápidos en su posición. En los Anexos B.1, B.2 y B.3 se muestran secuencias exitosas del seguimiento visual de este vídeo en la cámara 1, 2 y 3, respectivamente.

En las Tablas 4.3, 4.4 y 4.5, se observan los mejores resultados obtenidos del análisis del segundo vídeo para cada cámara, respectivamente. Las pruebas incluyen al igual que en el primer vídeo, el ruido sal y pimienta. Para esto, se analizaron tres pruebas para el espacio de color CIELab y dos para el espacio de color HSI, debido a que a un mayor nivel de ruido el sistema falla. El nivel de ruido base, al igual que el primer vídeo, corresponde a la adición del ruido sal y pimienta en cada cuadro, donde la probabilidad que un píxel sea blanco o negro es del 1 %. De igual manera, el ruido base x5, x10 y x20, representa que la probabilidad de que un pixel sea blanco o negro es del 5 %, 10 % y 20 %, respectivamente.

De manera general, en las Tablas 4.3, 4.4 y 4.5 se observan valores similares entre las métricas estudiadas. Esto se debe a que en este vídeo el objeto de interés posee una distribución de color visualmente homogénea. Además, el objeto es seleccionado inicialmente en la cámara 1 y en dichas tablas se evidencia que el seguimiento visual en las otras cámaras resulta bueno. De igual forma, se observa que el espacio de color HSI resulta menos robusto al ruido sal y pimienta que el espacio de color CIELab.

En las Tablas 4.6, 4.7 y 4.8 se muestran los errores calculados entre los parámetros obtenidos por el sistema y la verdad de referencia en cada cámara, de manera similar al primer vídeo. En estas tablas se observa que los espacios de color HSI y CIELab presentan desempeños similares, pero al igual que el vídeo anterior, cabe destacar que el ruido sal y pimienta afecta en mayor medida al sistema utilizando el espacio de color HSI.

El análisis de la frecuencia de procesamiento en este vídeo se muestra en la Tabla 4.9, donde se observa que los cálculos en el espacio de color HSI son más rápidos que en el espacio de color CIELab, como se observó en las pruebas del primer vídeo.

Tabla 4.3: *Mejores resultados del análisis del segundo vídeo para la primera cámara.*

CIELab			
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**
Sin ruido	0.861	0.859	0.855
Ruido base	0.874	0.851	0.857
Ruido base x5	0.909	0.828	0.862
HSI			
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**
Sin ruido	0.873	0.860	0.863
Ruido base	0.893	0.842	0.862

Tabla 4.4: *Mejores resultados del análisis del segundo vídeo para la segunda cámara.*

CIELab			
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**
Sin ruido	0.710	0.813	0.750
Ruido base	0.845	0.792	0.812
Ruido base x5	0.873	0.795	0.828
HSI			
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**
Sin ruido	0.740	0.838	0.781
Ruido base	0.735	0.762	0.744

Tabla 4.5: *Mejores resultados del análisis del segundo vídeo para la tercera cámara.*

CIELab			
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**
Sin ruido	0.843	0.888	0.860
Ruido base	0.858	0.883	0.866
Ruido base x5	0.873	0.862	0.864
HSI			
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**
Sin ruido	0.818	0.877	0.842
Ruido base	0.867	0.872	0.866

Tabla 4.6: *Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el segundo vídeo en la cámara 1.*

CIELab				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	4.57	6.90	24.71	39.42
Ruido base	4.69	5.95	27.07	32.81
Ruido base x5	3.99	3.43	50.20	21.25
HSI				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	3.69	5.18	22.70	29.80
Ruido base	3.96	4.31	27.69	21.37

Tabla 4.7: *Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el segundo vídeo en la cámara 2.*

CIELab				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	18.26	32.27	44.39	106.41
Ruido base	17.53	7.37	33.32	43.39
Ruido base x5	11.79	6.14	34.10	28.27
HSI				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	18.78	17.17	19.63	110.71
Ruido base	16.07	6.57	18.07	42.22

Tabla 4.8: *Análisis de la desviación estándar del error obtenido del centro en x, centro en y, en el ancho y alto del objeto en el segundo vídeo en la cámara 3.*

CIELab				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	2.62	9.87	12.50	66.71
Ruido base	2.72	8.63	13.03	53.89
Ruido base x5	3.17	5.16	14.18	30.05
HSI				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	3.81	13.57	17.99	58.52
Ruido base	3.34	4.93	15.58	42.28

Tabla 4.9: *Análisis de la frecuencia de operación del segundo vídeo.*

CIELab	
Nivel de Ruido	**Frecuencia (Hz)**
Sin ruido	32.58
Ruido base	34.68
Ruido base x5	34.66
HSI	
Nivel de Ruido	**Frecuencia (Hz)**
Sin ruido	42.91
Ruido base	40.49

Selección de información

En el tercer vídeo se realiza el seguimiento de un carro a escala de color verde sobre una pista ovalada vista desde tres cámaras, con la finalidad de evaluar el

desempeño de la elección de la cámara con mejor vista del objeto. En este vídeo los cambios de iluminación son semi-controlados y el objeto presenta movimientos rápidos y cambios repentinos en su posición. Dicho vídeo fue grabado en una escena de interior y la calibración del sistema se realizó como se mencionó en el capitulo anterior. Además, la verdad de referencia de cada cámara fue calculada manualmente. Estas cámaras están distribuidas espacialmente como se observa en la Figura 4.2. Donde, dichas cámaras se encuentran ubicadas en las coordenadas (252.0,-2030.2,729.7), (1683.0,360.5,360.4) y (-1037.0,-3.6,413.0) respectivamente, en milímetros, teniendo como referencia el marco de coordenadas global. En los Anexos C.1, C.2 y C.3 se muestran secuencias exitosas del seguimiento visual de este vídeo en la cámara 1, 2 y 3 respectivamente.

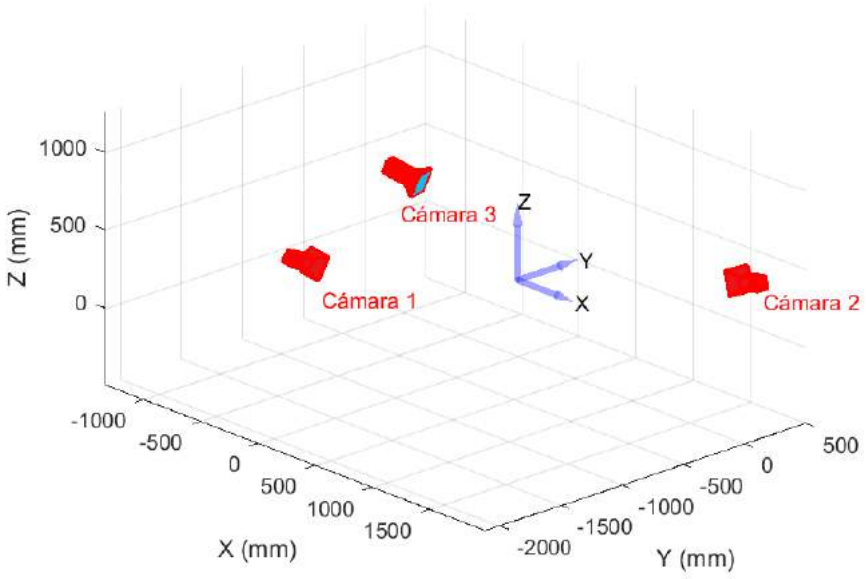

Figura 4.2: *Distribución espacial de las cámaras en el escenario del tercer vídeo.*

La evaluación del módulo de la selección de la cámara con mejor vista del objeto, fue realizada a través de un conjunto de sujetos de pruebas, con la finalidad de estimar dicha selección de manera natural por parte de los mismos. Para ello, un total de cinco personas seleccionaron manualmente para cada cuadro, la cámara que representa la mejor vista del objeto de interés, teniendo para esto tres opciones posibles, la cámara número 1, número 2 o número 3.

Una vez obtenidos los resultados de la selección de la cámara con mejor vista del objeto por parte de los sujetos de prueba, se analizaron dichos resultados evaluando por cada sujeto el error global (EG) entre el mismo y la selección automática del sistema, utilizando la Ecuación 4.5.

$$EG = 1 - \frac{1}{n}\sum_{i=1}^{n}(S_p^i \cap S_s^i) \tag{4.5}$$

donde, n representa el número total de cuadros estudiados, y S_p y S_s representan la selección de la cámara por parte del sujeto y el sistema respectivamente, tomando valores de 1, 2 o 3. Dicha ecuación compara en cada cuadro si la cámara seleccionada por el sujeto es igual a la cámara seleccionada por el sistema. Si la selección es igual, se obtiene un valor de 1, en caso contrario es 0. Finalmente, se obtiene un error global en el rango de 0 a 1 promediando dichos valores y restándolo a 1, donde el valor de 1 en dicho error corresponde a la máxima discordancia y 0 corresponde a la máxima concordancia en la selección.

Como se mencionó en el capítulo anterior, las características utilizadas para seleccionar la cámara con mejor vista del objeto es el área visual del objeto y la distancia espacial del mismo a cada cámara. Para combinar estas características se utilizó la constante de proporción α (ver Ecuación 3.14), cuyo valor fue estimado utilizando los resultados obtenidos por los sujetos de prueba. Para ello, se analizó el error global (ver Ecuación 4.5) para diferentes valores de la constante proporción, donde estos valores abarcan el rango de 0 a 1 en pasos de 0.01. En la Figura 4.3 se observa el error global para cada valor de la constante de proporción mencionada, para los cinco sujetos de prueba.

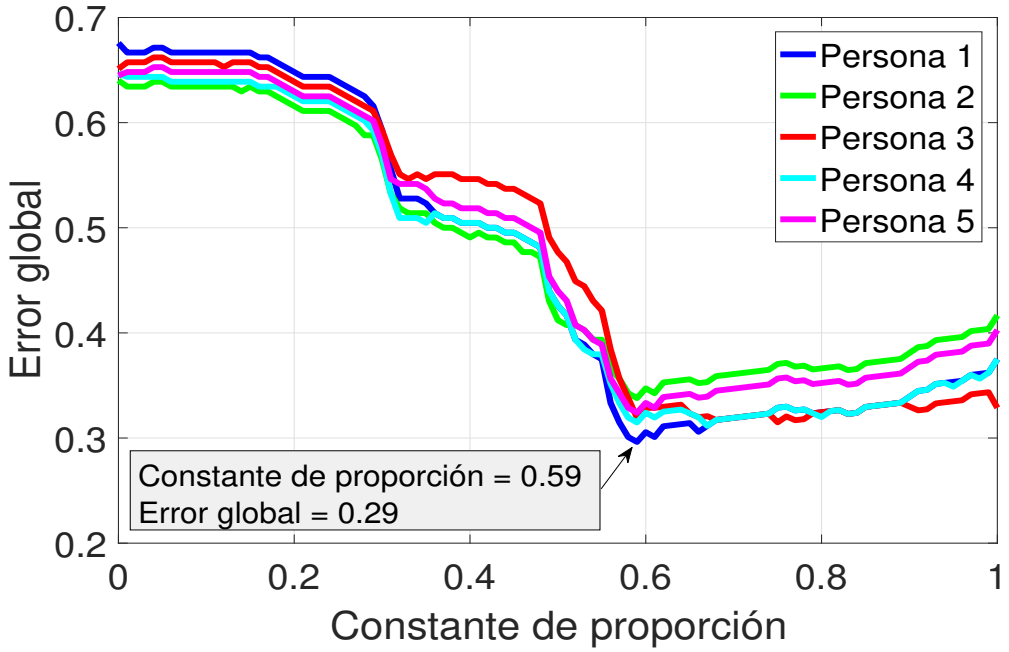

Figura 4.3: *Comportamiento del error global para cada valor de la constante de proporción, por cada sujeto de prueba.*

En la Figura 4.3 se observa que el valor de la constante de proporción promedio

que genera menor error global es de 0.59. Por lo que, dicho valor demuestra que el uso de una sola característica no es suficiente para estimar de manera adecuada la cámara con mejor vista del objeto. Este valor es utilizado por el sistema para seleccionar automáticamente la cámara con mejor vista del objeto, con la finalidad de analizar el desempeño del módulo estudiado en esta sección. En la Figura 4.4 se muestra la selección automática de la cámara con mejor vista del objeto para cada cuadro del vídeo.

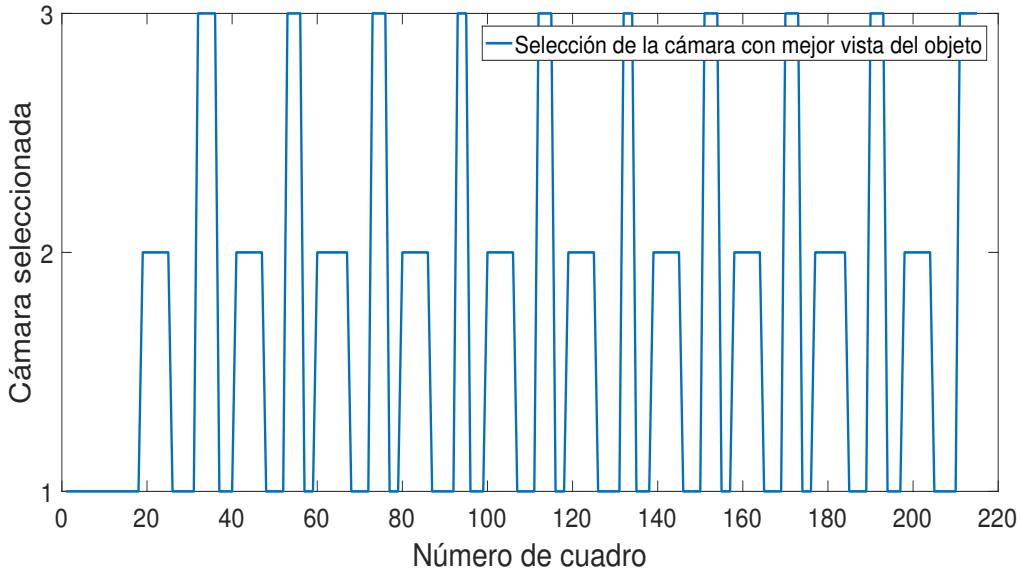

Figura 4.4: *Selección automática de la cámara con mejor vista del objeto.*

En la Tabla 4.10 se observa los mejores resultados obtenidos del análisis de la cámara con mejor vista del objeto en el tercer vídeo. Las pruebas incluyen al igual que en los vídeos anteriores, el ruido sal y pimienta. Para esto, se analizaron tres pruebas para el espacio de color CIELab, debido a que a un mayor nivel de ruido el sistema falla. Además, en este vídeo el espacio de color HSI falla en la identificación del objeto en las otras cámaras. En la Tabla 4.10 se refleja que el seguimiento del objeto fue realizado con buenos resultados a frecuencias relativamente altas.

Tabla 4.10: *Mejores resultados del análisis del tercer vídeo para la cámara seleccionada automáticamente.*

CIELab				
Nivel de Ruido	**Exhaustividad**	**Precisión**	**Métrica F**	**Frecuencia (Hz)**
Sin ruido	0.855	0.795	0.815	20.1
Ruido base	0.866	0.798	0.821	19.6
Ruido base x5	0.879	0.819	0.839	19.5

Los errores calculados entre los parámetros obtenidos por el sistema y la verdad

de referencia en la cámara con mejor vista del objeto, se observan en la Tabla 4.11. Los errores asociados al ancho y alto del objeto presentan mayor error, esto se debe a que el objeto se mueve rápidamente, produciendo un efecto de emborronamiento en el mismo, afectando así negativamente el desempeño del seguimiento.

Tabla 4.11: *Análisis de la desviación estándar del error obtenido del tercer vídeo en la cámara con mejor vista del objeto.*

CIELab				
Nivel de Ruido	**Centro en x**	**Centro en y**	**Ancho**	**Alto**
Sin ruido	32.49	28.69	144.49	120.40
Ruido base	31.76	28.15	141.28	118.91
Ruido base x5	19.36	26.63	78.89	114.80

En el Anexo C.4 se muestra una secuencia exitosa del seguimiento visual del objeto de interés en todas las cámaras, incluyendo la cámara con mejor vista del objeto.

4.4. Comparación con Otros Métodos

Finalmente, se realiza la comparación con el estado del arte. Sin embargo, dicha comparación se realiza únicamente con el primer vídeo analizado, donde se hace el seguimiento de una taza de color azul. Esto se debe a que muchos de los sistemas de seguimiento continuo de objetos en múltiples cámaras, utilizan objetos complejos que no pueden ser caracterizados únicamente por un color. Por ejemplo, el seguimiento de una persona en un paso peatonal, debido a que la persona puede poseer colores similares o idénticos a otra persona o al escenario, afectando negativamente el desempeño del sistema.

La Tabla 4.12 muestra una comparación del mejor resultado obtenido en el seguimiento visual de una taza de color azul, con métodos del estado del arte. Los valores de las métricas mostradas en dicha Tabla son las registradas en las referencias usando la base de datos BoBot, mencionada anteriormente. Obsérvese en esta tabla que el método utilizado, genera un desempeño superior a otros.

Los métodos MSA-T[12], SOAMST[25] y KMS[20], utilizan el color del objeto como característica principal de información y se basan en el algoritmo de $mean\ shift$; los métodos DFT[29] y MTT[33] se basan en el seguimiento de la distribución de probabilidades de los píxeles en la imagen. Considerando lo anterior, se observa que el color es una de las características más utilizadas para modelar los objetos en los sistemas para el seguimiento continuo de objetos, obteniendo buenos desempeños.

Tabla 4.12: *Comparación del estado del arte y el sistema desarrollado para el primer vídeo analizado.*

Exhaustividad	Precisión	Métrica F	Método
0.91	0.94	0.93	MSA-T [12]
0.86	0.89	0.87	SOAMST [25]
0.70	0.78	0.75	DFT [29]
0.54	**0.87**	**0.65**	**Desarrollado**
0.52	0.59	0.57	KMS [20]
0.39	0.46	0.42	MTT [33]

4.5. Conclusiones del Capítulo

En este capítulo se explicó el protocolo de pruebas que se siguió para evaluar el desempeño del sistema desarrollado, donde se evaluaron distintos vídeos con diferentes características, para así determinar el alcance del sistema. En cada prueba se hizo la comparación de los resultados obtenidos usando el espacio de color HSI y el espacio de color CIELab para determinar sus desempeños en el sistema.

Se concluyó que el desempeño del sistema utilizando los espacios de color HSI y CIELab son similares, aunque el tiempo de procesamiento es menor en el espacio de color HSI. Sin embargo, el espacio de color CIELab es más robusto cuando en el escenario se presentan colores similares al objeto de interés, y al ruido del tipo sal y pimienta. Finalmente, cabe destacar que como el sistema desarrollado utiliza únicamente la información de color del objeto, el seguimiento falla si el escenario posee colores muy similares al del objeto de interés, sin importar el espacio de color.

Capítulo 5

Conclusiones y Perspectivas

En este capítulo se mencionan los aspectos relevantes que surgieron del desarrollo de este proyecto, siendo parte de éstos, los aprendizajes obtenidos y las contribuciones realizadas. Además, también se mencionan algunas mejoras que se le pueden realizar al sistema desarrollado.

Conclusiones

En este proyecto de tesis, se desarrolló un sistema para el seguimiento continuo de objetos en una red de cámaras modeladas, con campos de visión compartida. El sistema inicialmente requiere seleccionar una zona de color homogéneo del objeto de interés en una cámara. Posteriormente, el sistema identifica dicho objeto en las otras cámaras utilizando la información de color del mismo, a través de las relaciones geométricas entre las mismas. Una vez identificado el objeto, se realiza el seguimiento continuo de éste en cada cámara de manera independiente. Además, el sistema selecciona y despliega continuamente la imagen de la cámara con mejor vista del objeto, utilizando su información espacial y el área que ocupa en la imagen.

El sistema fue probado utilizando tres vídeos diferentes en entornos de interior y exterior. Esto permitió comprobar en primera instancia la buena efectividad en el seguimiento del objeto de interés, a pesar de que las imágenes presentan perturbaciones en sus píxeles, causadas por lo cambios en la iluminación. Sin embargo, cuando existen en el escenario zonas de color muy similares al color del objeto, el sistema puede presentar fallas debido a la confusión de dicha zona con el objeto.

En las pruebas realizadas al sistema se analizaron dos espacios de color perceptual, siendo éstos, el espacio de color CIELab y HSI. Dichas pruebas dieron

como resultado que dichos espacios de color presentan desempeños similares. Sin embargo, cuando el escenario posee regiones de color similar al objeto de interés y las imágenes presentan ruido del tipo sal y pimienta, el espacio de color CIELab es más robusto. Además, en el espacio de color HSI es menor el tiempo de cómputo que en el espacio de color CIELab.

La mayor contribución realizada en este proyecto, corresponde a la integración de diferentes técnicas para dar solución al problema planteado. Adicionalmente, se propuso un módulo con la finalidad de seleccionar continuamente la cámara con mejor vista del objeto, a partir de la información espacial y el área visual del objeto. Este módulo permite estimar dicha selección tratando de seguir la selección natural de una persona, tal como fue demostrado con los resultados obtenidos en las pruebas.

De forma general, los resultados obtenidos al probar el sistema en escenarios de interior y exterior, son buenos. Además, la frecuencia de procesamiento requerido para el seguimiento visual de los objetos en múltiples cámaras es suficientemente alta para que el sistema funcione en tiempo real, dependiendo del tamaño visual de los objetos y el número de cámaras presentes en el escenario. Benfold y Reid [1] señalan que una frecuencia mayor a los 25 Hz, se considera una velocidad en tiempo real.

Perspectivas

En este trabajo de tesis se presentaron diversos planteamientos para la resolución del problema en estudio. Durante la realización de este trabajo se desarrollaron habilidades y conocimientos para la elaboración y diseño de los sistemas propuestos. Además, se realizó la investigación y análisis de los métodos existentes relacionados a este tema.

El sistema propuesto fue dividido en dos etapas principales: El seguimiento visual de objetos y la fusión de información. Esto resulta interesante puesto que en la primera etapa pueden ser usadas diferentes técnicas y modelos de objetos según la aplicación lo requiera, y en la segunda etapa, fusionar la información disponible, permitiendo mantener la estructura planteada.

En este trabajo se utilizó únicamente la información de color del objeto para modelarlo, generando fallas cuando el sistema confunde dicho objeto con zonas del escenario. Estas fallas podrían reducirse utilizando otras características como la forma, textura e información temporal del objeto. Sin embargo, el tiempo de cómputo debe ser considerado para no aumentarlo en gran medida.

La aplicación de los módulos propuestos, pueden ser ajustados dependiendo de las características del objeto y el conocimiento previo del escenario. Por ejemplo, si se desea seguir autos en un estacionamiento visto desde uno o más cámaras, la forma del auto puede ser considerada para discernir el mismo de otros objetos.

Anexos

Anexo A

Seguimiento Visual de Objetos

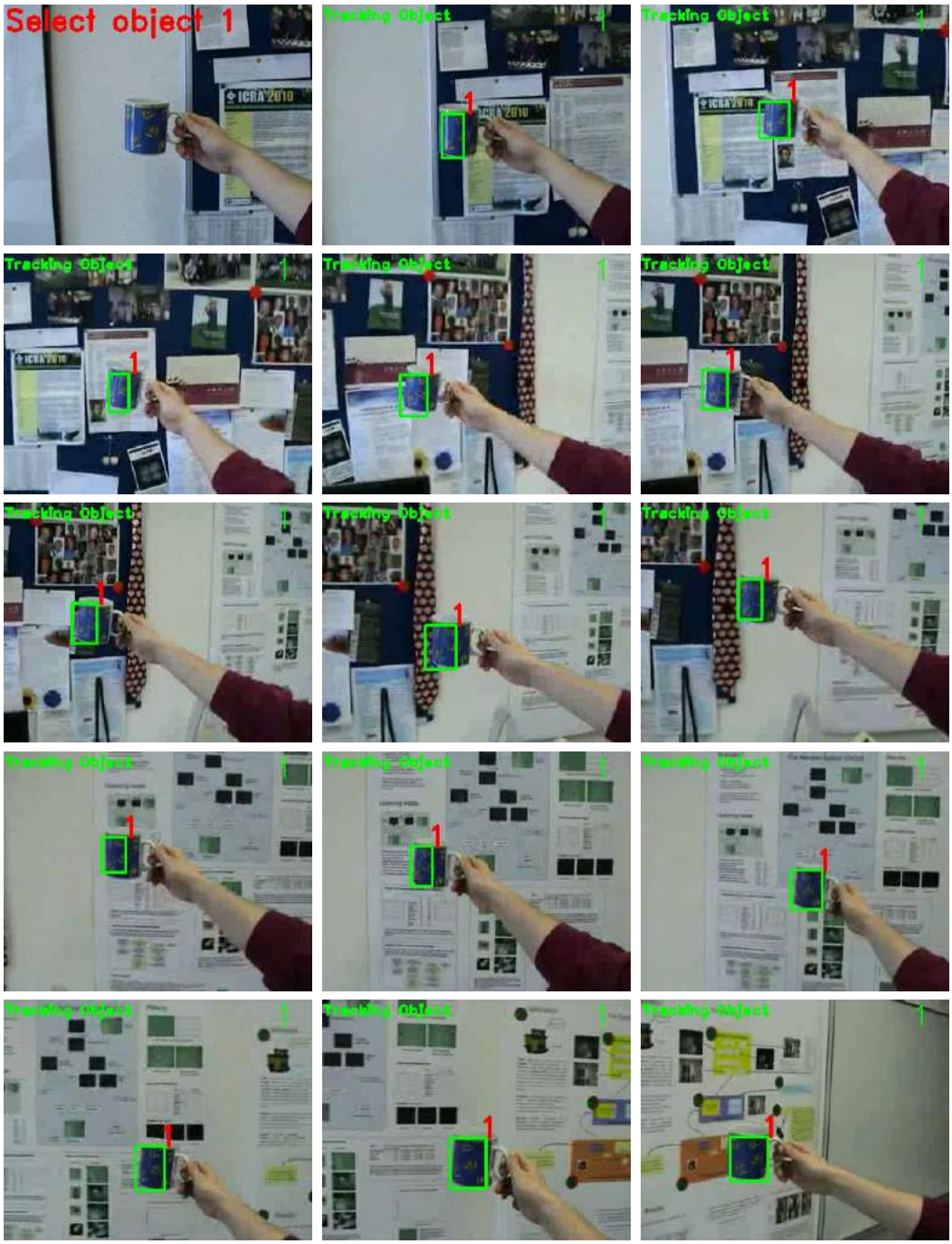

Figura A.1: *Seguimiento visual de una taza azul en un ambiente semi-controlado.*

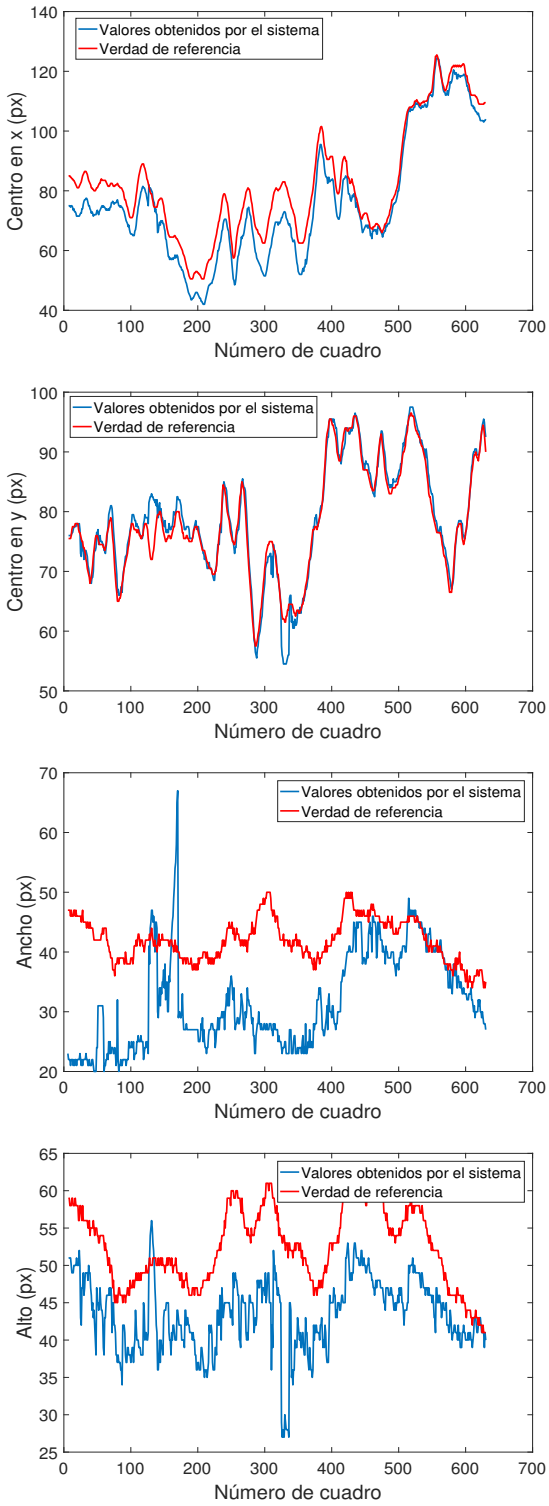

Figura A.2: *Gráficas entre la verdad de referencia y los valores calculados por el sistema para cada cuadro del vídeo uno.*

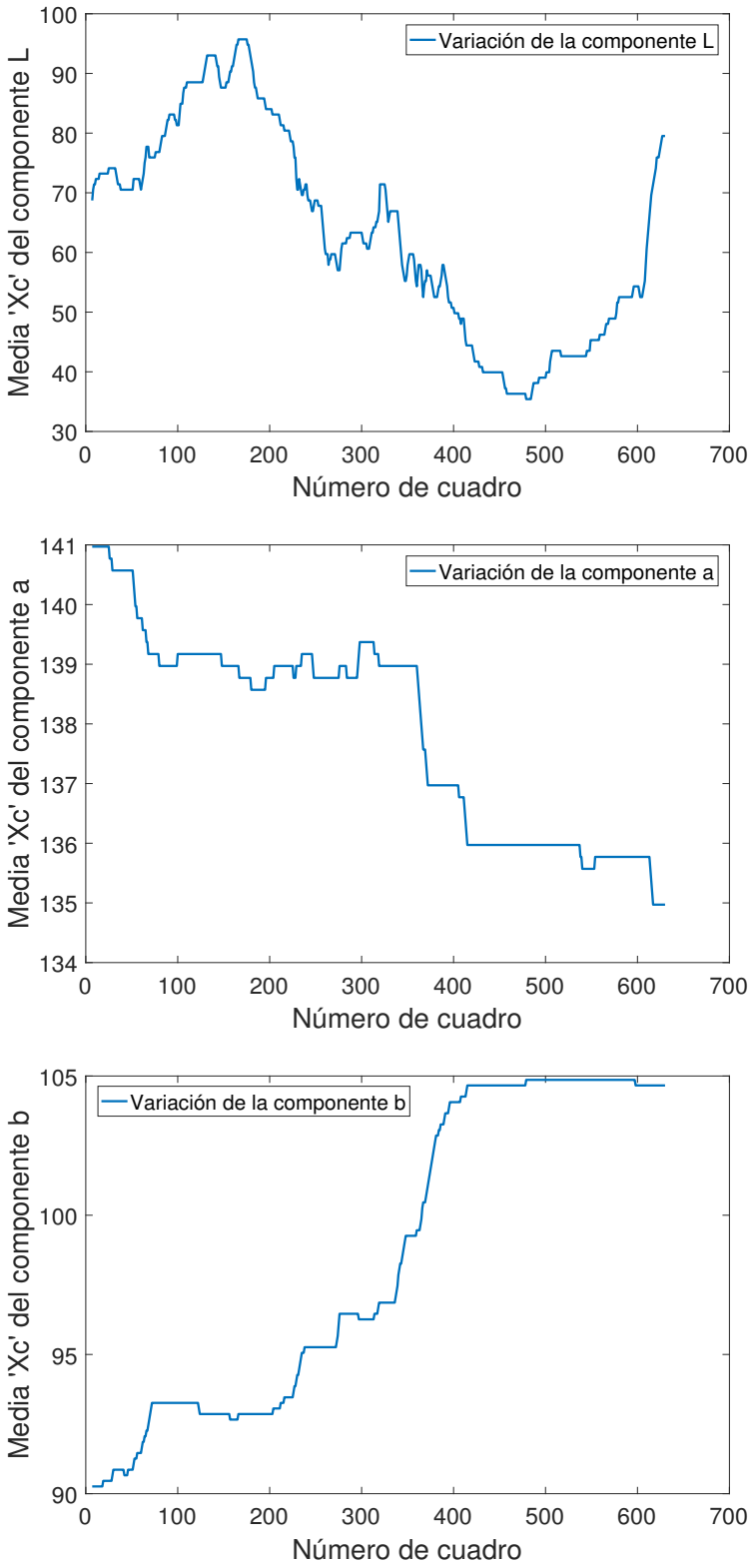

Figura A.3: *Gráficas de la variación dinámica de las componentes del espacio de color CIELab para cada cuadro del vídeo uno.*

Anexo B

Agregación de la Información Disponible

Figura B.1: *Seguimiento visual de un suéter azul en un ambiente no controlado visto desde la cámara 1 en el vídeo dos.*

Figura B.2: *Seguimiento visual de un suéter azul en un ambiente no controlado visto desde la cámara 2 en el vídeo dos.*

Figura B.3: *Seguimiento visual de un suéter azul en un ambiente no controlado visto desde la cámara 3 en el vídeo dos.*

Anexo C

Selección de Información

Figura C.1: *Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara 1 en el vídeo tres.*

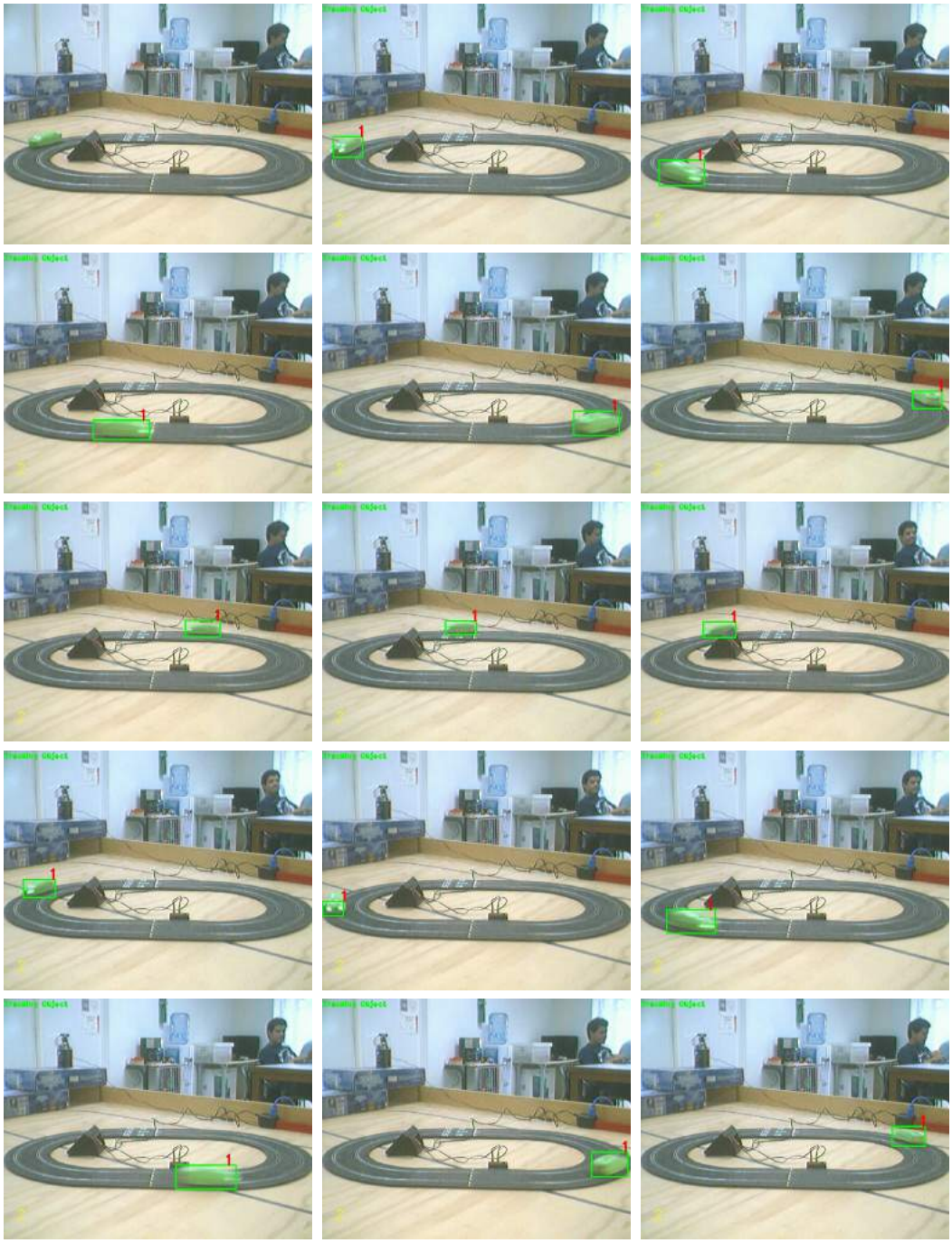

Figura C.2: *Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara 2 en el vídeo tres.*

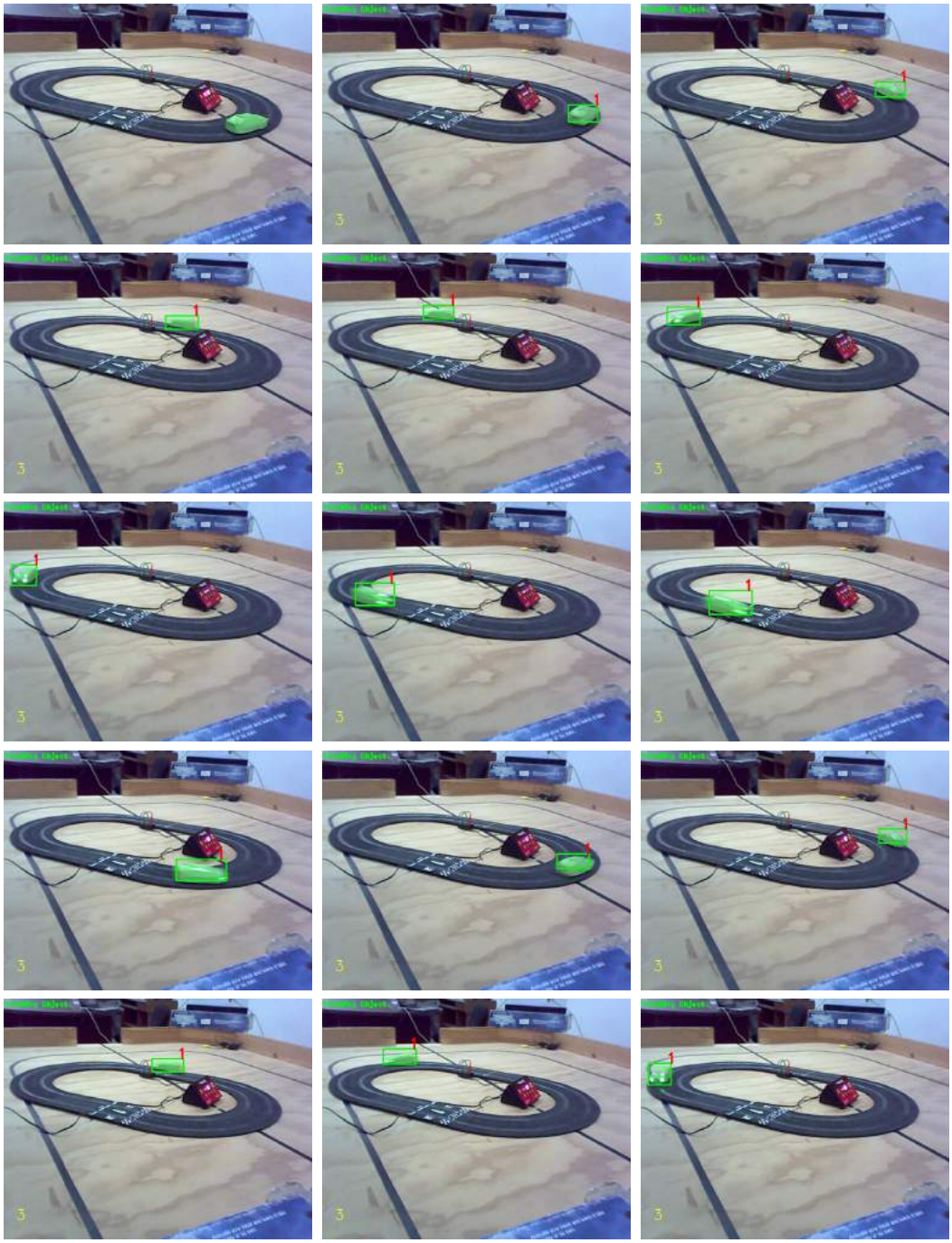

Figura C.3: *Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara 3 en el vídeo tres.*

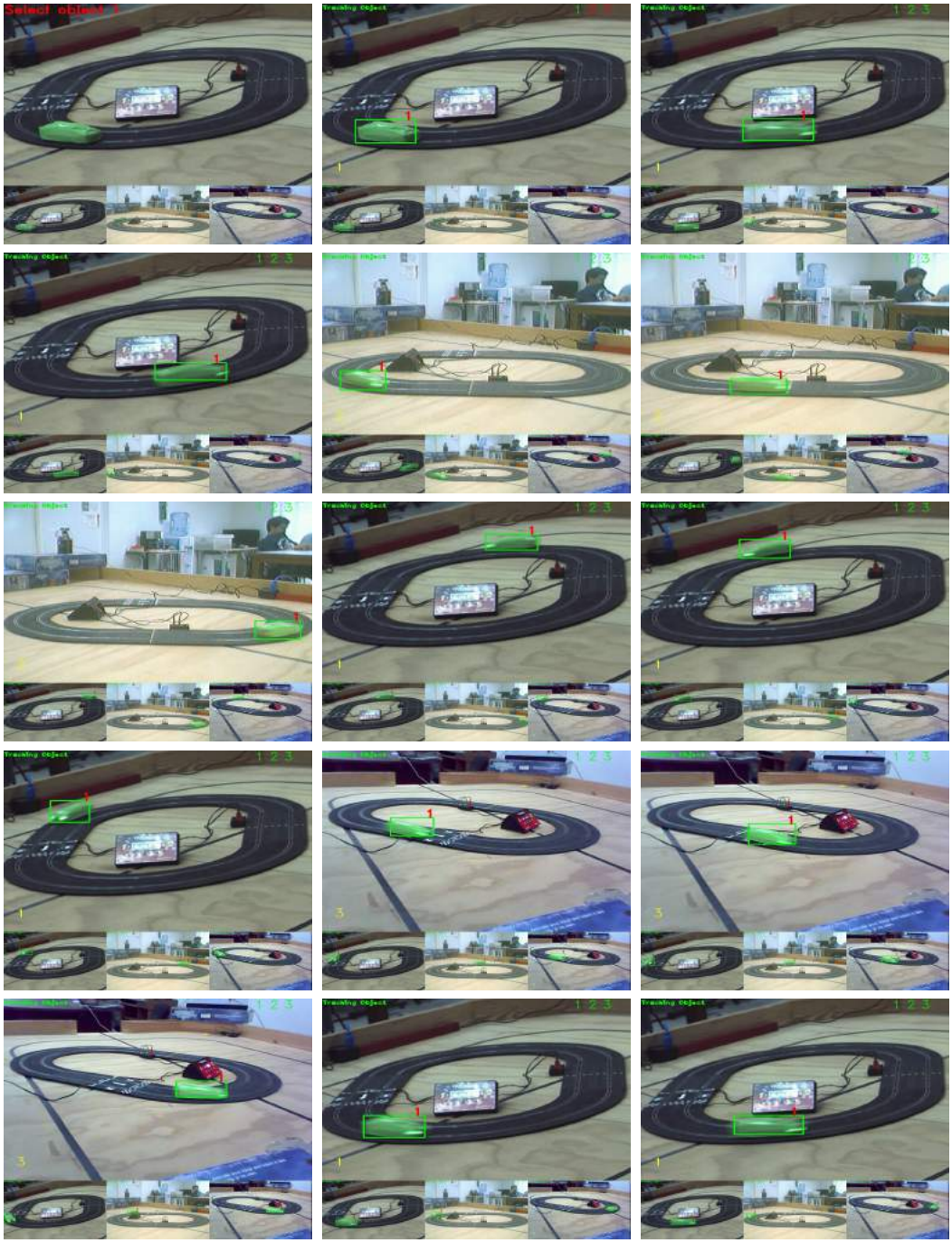

Figura C.4: *Seguimiento visual de un carro a escala color verde en un ambiente semi-controlado visto desde la cámara con mejor vista del objeto en el vídeo tres.*

Bibliografía

[1] Benfold, B. and Reid, I. (2011). Stable Multi-Target Tracking in Real-Time Surveillance Video. *IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 3457–3464.

[2] Bhattacharyya, A. (1946). On a Measure of Divergence between Two Multinomial Populations. *Sankhyā Indian J. Stat.*, 7(4):401–406.

[3] Bouguet, J. Y. (2004). Camera Calibration Toolbox for Matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/`. Accesed: December, 2017.

[4] Bredereck, M., Xiaoyan, J., Korner, M., and Denzler, J. (2012). Data association for multi-object Tracking-by-Detection in multi-camera networks. In *Sixth Int. Conf. Distrib. Smart Cameras (ICDSC), Hong Kong*, pages 1–6. IEEE.

[5] Cai, Y., Chen, W., Huang, K., and Tan, T. (2007). Continuously Tracking Objects Across Multiple Widely Separated Cameras. In *Comput. Vis. – ACCV 2007*, volume 4843, pages 843–852. Springer Berlin Heidelberg, Berlin, Heidelberg.

[6] Cai, Z., Hu, S., Shi, Y., Wang, Q., and Zhang, D. (2017). Multiple human tracking based on distributed collaborative cameras. *Multimed. Tools Appl.*, 76(2):1941–1957.

[7] Chao, G.-C., Jeng, S.-K., and Lee, S.-S. (2011). An improved occlusion handling for appearance-based tracking. In *2011 18th IEEE Int. Conf. Image Process.*, pages 465–468. IEEE.

[8] Chen, Y., Zhao, Q., An, Z., Lv, P., and Zhao, L. (2016). Distributed Multi-Target Tracking Based on the K-MTSCF Algorithm in Camera Networks. *IEEE Sens. J.*, 16(13):5481–5490.

[9] Chen, Z., Liao, W., Xu, B., Liu, H., Li, Q., Li, H., Xiao, C., Zhang, H., Li, Y., Bao, W., and Yang, D. (2015). Object Tracking over a Multiple-Camera Network. In *2015 IEEE Int. Conf. Multimed. Big Data*, pages 276–279. IEEE.

[10] Chui, C. K. and Chen, G. (2009). *Kalman Filtering*. (4th. ed.) Berlin: Springer Berlin Heidelberg.

[11] Comaniciu, D., Ramesh, V., and Meer, P. (2000). Real-time tracking of non-rigid objects using mean shift. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognition. CVPR 2000 (Cat. No.PR00662)*, volume 2, pages 142–149. IEEE Comput. Soc.

[12] Ebrahim Shiri, M., Fatemeh Razavi, S., and Sajedi, H. (2016). Integration of colour and uniform interlaced derivative patterns for object tracking. *IET Image Process.*, 10(5):381–390.

[13] Elhayek, A., de Aguiar, E., Jain, A., Tompson, J., Pishchulin, L., Andriluka, M., Bregler, C., Schiele, B., and Theobalt, C. (2015). Efficient ConvNet-based marker-less motion capture in general scenes with a low number of cameras. In *2015 IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 3810–3818. IEEE.

[14] Farnebäck, G. (2003). Two-Frame Motion Estimation Based on Polynomial Expansion. In *Image Anal. 13th Scand. Conf. SCIA 2003 Halmstad, Sweden, June 29 – July 2, 2003 Proc.*, number 1, pages 363–370.

[15] Hartley, R. I. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. (2nd. ed.) New York: Cambridge Univ. Press.

[16] Hui-Huang, H., Wei-Min, Y., and Shih, T. K. (2013). People tracking in a multi-camera environment. In *IEEE Conf. Anthol.*, pages 1–4. IEEE.

[17] Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *J. Basic Eng.*, 82(1):35.

[18] Khan, S. and Shah, M. (2003). Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(10):1355–1360.

[19] Li, M., Jiang, Z., Tang, J., and Zhao, C. (2015). Activity Prediction Based on Spatiotemporal Model in a Multiple Cameras Network. In *2015 IEEE Int. Conf. Multimed. Big Data*, pages 272–275. IEEE.

[20] Li, X., Zhang, T., Shen, X., and Sun, J. (2010). Object tracking using an adaptive Kalman filter combined with mean shift. *Opt. Eng.*, 49(2):49–51.

[21] Meyer, F. (1992). Color Image Segmentation. In *1992 Int. Conf. Image Process. its Appl.*, pages 303–306.

[22] Montecillo, P. (2003). *Sistema de Seguimiento de Objetos en Tiempo Real Mediante Caracterización Difusa del Color*. Tesis de licenciatura, Universidad de Guanajuato, Guanajuato.

[23] Montecillo, P. (2006). *Seguimiento de Objetos Rigidos y Articulados Representados por Color Difuso para su Aplicación en Visión Robótica.* Tesis de maestría, Universidad de Guanajuato, Guanajuato.

[24] Narkhede, P. R. and Gokhale, A. V. (2015). Color particle filter based object tracking using frame segmentation in CIELab* and HSV color spaces. In *2015 Int. Conf. Commun. Signal Process.*, pages 0804–0808. IEEE.

[25] Ning, J., Zhang, L., Zhang, D., and Wu, C. (2012). Scale and orientation adaptive mean shift tracking. *IET Comput. Vis.*, 6(1):52.

[26] Pan, J. and Hu, B. (2007). Robust Occlusion Handling in Object Tracking. In *2007 IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1–8. IEEE.

[27] Papadakis, N. and Bugeau, A. (2011). Tracking with Occlusions via Graph Cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(1):144–157.

[28] Powers, D. M. W. (2007). Evaluation : From Precision , Recall and F-Factor to ROC , Informedness , Markedness & Correlation. *Mach. Learn. Technol.*, 2(1):37–63.

[29] Sevilla-Lara, L. and Learned-Miller, E. (2012). Distribution fields for tracking. In *2012 IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 1910–1917. IEEE.

[30] Shen, W., Wu, Y., and Jia, Y. (2017). Compact discriminative object representation via weakly supervised learning for real-time visual tracking. *IET Comput. Vis.*, (iii):585–595.

[31] Smeulders, A. W., Chu, D. M., Cucchiara, R., Calderara, S., Dehghan, A., and Shah, M. (2014). Visual tracking: An experimental survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(7):1442–1468.

[32] Suha Kwak, Woonhyun Nam, Bohyung Han, and Joon Hee Han (2011). Learning occlusion with likelihoods for visual tracking. In *2011 Int. Conf. Comput. Vis.*, pages 1551–1558. IEEE.

[33] Tianzhu Zhang, Ghanem, B., Si Liu, and Ahuja, N. (2012). Robust visual tracking via multi-task sparse learning. In *2012 IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 2042–2049. IEEE.

[34] Vojíř, T. (2017). Tracking Dataset. `http://cmp.felk.cvut.cz/~vojirtom/dataset/tv77/`. Accessed: December, 2017.

[35] Wang, H., Kläser, A., Schmid, C., and Liu, C.-L. (2013). Dense Trajectories and Motion Boundary Descriptors for Action Recognition. *Int. J. Comput. Vis.*, 103(1):60–79.

[36] Welch, G. and Bishop, G. (2006). An Introduction to the Kalman Filter. *In Pract.*, 7(1):1–16.

[37] Yilmaz, A., Javed, O., and Shah, M. (2006). Object tracking. *ACM Comput. Surv.*, 38(4):13–58.

[38] Yun, Y., Gu, I., and Aghajan, H. (2012). Maximum-likelihood object tracking from multi-view video by combining homography and epipolar constraints. *Distrib. Smart Cameras (ICDSC), 2012 Sixth Int. Conf.*, pages 1 – 6.

[39] Yun, Y., Gu, I. Y.-H., Provost, J., and Akesson, K. (2013). Multi-view hand tracking using epipolar geometry-based consistent labeling for an industrial application. In *2013 Seventh Int. Conf. Distrib. Smart Cameras*, pages 1–6. IEEE.