



Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias de la Computación

**Fusión de Datos Para el Seguimiento de Personas
Empleando Tres Algoritmos de Asociación de Datos
NNKF, PDA y JPDA**

Tesis

Que para Obtener el Título de:

Ingeniero en Ciencias de la Computación

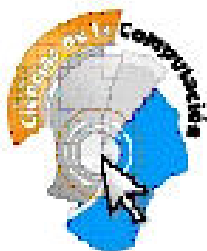
P R E S E N T A:

Jesús Ernesto Uribe Hernández

TUTOR:

Manuel Martín Ortiz

Septiembre 2010



PREFACIO

El objetivo principal de esta tesis es desarrollar un sistema de fusión a nivel características para la detección y seguimiento de personas en un escenario controlado, empleando los algoritmos de asociación NNKF, PDA y JPDA.

El Filtro Kalman, (indicado como KF, acrónimo en inglés Kalman Filter), es un filtro que provee una solución recursiva eficiente del método de mínimos cuadrados. Este algoritmo tiene un conjunto de ecuaciones matemáticas que permiten calcular un estimador lineal de un proceso con base en la información disponible de cada momento de tiempo $t - 1$ y actualiza las estimaciones en el tiempo t . La importancia del Filtro Kalman radica en la aplicación que se puede hacer en los tres algoritmos mencionados anteriormente.

NNKF asocia probabilidades de medidas cercanas a un solo objetivo, ya que trabaja de manera similar a Nearest Neighbor. PDA calcula la asociación de probabilidades por cada medida válida en el tiempo actual de un objetivo de interés. Esta información de probabilidad es empleada por el filtro de estimación, generada en una iteración de tiempo, si alguna de las medidas cae dentro del umbral, dado por PDA, se asume entonces que un objetivo ha sido inicializado al evento de asociación, las medidas tienen que ser mutuamente exclusivas y exhaustivas. JPDA es una extensión del PDA, la diferencia radica en que es el único de los tres algoritmos diseñado para asociar medidas de distintos objetivos de interés.

El desarrollo del código ha sido sobre la plataforma .NET, apartir del lenguaje de programación C# (C sharp), realizado en el ambiente de desarrollo de Visual Studio, además de emplear la máquina de visión Halcon de MVTec (MVTec software GmbH) y las librerías de ésta, para el desarrollo de la aplicación visual. El procesador utilizado fué un Pentium IV a 2.2 Ghz.

Las secuencias de imágenes analizadas, fueron tomadas en un pasillo de un centro comercial, donde el paso de personas es continuo, estas secuencias pertenecen al proyecto CAVIAR. El propósito es dar seguimiento a los individuos que pasan sobre este pasillo, identificando las coordenadas (x, y) de cada region de interés, generando una matriz homográfica y finalmente asociar los datos para dar seguimiento a las personas.

Esta tesis se desarrolló en gran parte en el Laboratorio de Visión de la Coordinación de Ciencias Computacionales del Instituto Nacional de Astrofísica, Óptica y Electrónica, bajo la supervisión de la Dra. Janeth Cruz Enriquez y de el co - asesor Dr. Manuel Martín Ortiz de la Facultad de Ciencias de la Computación, de la Benemérita Universidad Autónoma de Puebla.

Resumen

Para los seres vivos y en especial para los seres humanos, la manera de comprender eventos en nuestro entorno, considerados agradables o desagradables, se realiza mediante la combinación de lo que los sentidos perciben, donde la toma de decisiones es sumamente importante.

El sentido de la vista, para este caso en particular, proporciona mucho más información que quizás cualquier otro, ya que uno puede, de cierto modo, entender lo que está sucediendo en lo que se está observando y el cerebro nos permite, en algunos casos, predecir la acción que posiblemente pueda ocurrir a partir de los sucesos previamente observados.

Así como se adquiere esta información en los seres vivos, en el área computacional existen distintas formas de obtener datos, combinarlos y tomar decisiones. Estos se pueden adquirir usando sensores de sonido, tacto, luz, temperatura, imagen, etc. Una vez que se tienen tales muestras, es necesario el análisis a partir de algoritmos de fusión de datos, los cuales tienen como objetivo realizar una combinación óptima de la información. Algunos de estos algoritmos se basan en diferentes técnicas, entre ellas se encuentran el Nearest Neighbor, Fuzzy Logic, PDA, JPDA, Redes Neuronales, Inferencia Bayesiana o Teoría Evidencial (DST ó DSMT), etc.

El hablar de fusión en sensores de imagen, se refiere a utilizar varias cámaras digitales enfocadas hacia algún escenario en común, esto permite obtener información de un mismo centro de proyección. Sin embargo, la exactitud que se tenga en los datos recibidos por una de las cámaras, no puede ser del todo precisa y se corre el riesgo de perder información, es por ello que se requiere además de múltiples cámaras, ya sea visibles o infrarrojas, una arquitectura cooperativa o complementaria con el fin de obtener datos con menos errores y generar mejores resultados en la fusión de datos.

Índice general

Índice de Figuras	IV
Índice de Tablas	VI
1. Introducción	2
1.1. Planteamiento del Problema	3
1.2. Objetivo General	5
1.3. Objetivos específicos	5
1.4. Organización de la tesis	5
2. Fundamento Teórico	7
2.1. Fusión de Datos	7
2.1.1. Aplicaciones de los modelos multisensor	7
2.1.2. Ventajas de sistemas multisensores	9
2.2. Modelo de Fusión de Datos	10
2.2.1. Modelo JDL (Joint Directors of Laboratories):	11
2.3. Fusión en Imágenes	13
2.4. Segmentación	16
2.4.1. Métodos de Segmentación	16
2.4.2. Problemas de la segmentación	17
2.5. Geometría Proyectiva	18
2.5.1. Transformación Proyectiva	19
2.5.2. Determinación de la Matriz Proyectiva Homogénea 2D	21
2.6. Estimación, Decisión y Seguimiento con Múltiples Sensores	21
2.6.1. Seguimiento con múltiples sensores	23
2.6.2. Ecuaciones para múltiples sensores	23
2.6.3. Actualización Secuencial	24
2.6.4. Filtro Kalman	25
2.7. Asociación	26
2.7.1. Región de Validación	28
2.7.2. Problemas de la Asociación: Un Objetivo en Interferencia	28
2.7.3. Problemas de la Asociación: Múltiples Objetivos en Interferencia	29
2.7.4. Modelo del Objetivo	30

2.7.5.	Modelo para el Seguimiento de un solo Objetivo en Interferencia	31
2.7.6.	Algoritmos de Seguimiento para un solo Objetivo en Interferencia	32
2.7.6.1.	Algoritmo No Bayesiano: Nearest Neighbor Kalman Filter	32
2.7.6.2.	Algoritmo Bayesiano: Probabilistic Data Association	32
2.7.6.2.1.	Enfoque PDA	33
2.7.6.2.2.	Medidas de Validación	33
2.7.6.2.3.	El Estado de Estimación	34
2.7.6.2.4.	Estado de la Covarianza Actualizado	35
2.7.6.2.5.	Ecuaciones de Predicción	35
2.7.6.2.6.	Probabilistic Data Association	36
2.7.6.2.7.	PDA Paramétrico	37
2.7.7.	Algoritmo Bayesiano: Joint Probabilistic Data Association	37
2.7.7.0.8.	Enfoque JPDA	38
2.7.7.0.9.	Estado de estimación	38
2.7.7.0.10.	Características especiales del JPDA	39
2.7.7.0.11.	La Matriz de Validación	39
2.7.7.0.12.	JPDA Paramétrico	39

3.	Implementación de los Algoritmos de Fusión basados en la Asociación de Datos	41
3.1.	Introducción	41
3.2.	Modelo de Fusión	41
3.3.	Detección	42
3.3.1.	Proceso de la Segmentación	42
3.4.	Plano Homográfico	44
3.5.	Filtro Kalman	46
3.6.	NNKF	47
3.7.	PDA y JPDA	48
4.	Pruebas del Sistema de Seguimiento y Resultados	51
4.1.	Introducción	51
4.2.	Segmentación	51
4.3.	Transformación Proyectiva	53
4.4.	Pruebas con el Filtro Kalman Lineal	55
4.4.1.	Ecuación Lineal con el Filtro Kalman	55
4.4.2.	Secuencias de Imágenes con el Filtro Kalman	58
4.5.	Pruebas con el algoritmo NNKF	60
4.6.	Pruebas con el algoritmo PDA	62
4.7.	Pruebas con el algoritmo JPDA	64

5. Resultados	69
5.1. Introducción	69
5.2. Resultados en la Detección	69
5.3. Resultados en la Transformación Proyectiva	70
5.4. Resultados Filtro Kalman Lineal	71
5.5. Resultados y Comparaciones entre NNKF y PDA	72
5.5.1. Resultados NNKF	72
5.5.2. Resultados PDA	73
5.6. Comparaciones NNKF y PDA	74
5.7. Resultados JPDA	75
6. Conclusiones y Perspectivas	76
6.1. Conclusiones	76
6.2. Perspectivas	77
A. Modelos de Fusión de Datos	78
A.0.1. Modelo Ciclo de Inteligencia o IC, (The Intelligence Cycle): . . .	78
A.0.2. Modelo Waterfall (The Waterfall Model):	80
A.0.3. Modelo BCL, (The Boyd Control Loop or OODA Loop):	82
A.0.4. Modelo Dasarathy (The Dasarathy Model):	83
A.0.5. Modelo Omnibus (The Omnibus Model):	85
Appendices	78
B. Geometría proyectiva	87
B.0.6. Plano 2D	87
B.0.7. Puntos y líneas	87
B.0.8. Representación homogénea de puntos	88
B.0.9. Intersección de líneas	88
B.0.10. Puntos ideales y recta del infinito	88
B.1. Transformaciones del plano proyectivo	89
B.2. Transformación de líneas y cónicas	89
B.3. Jerarquía de transformaciones	90
B.3.1. Clase I: Isometrías	90
B.3.2. Clase II: Semejanzas	91
B.3.3. Clase III: Afinidades	92
B.3.4. Clase IV: Proyectividades	92
B.4. Descomposición de una proyectividad	93
C. Estimación, Decisión, Seguimiento y otros términos	94
D. Máquina de Visión Halcon	97
Referencias	98

Índice de Figuras

2.1.	Fusión de distintos tipos de sensores	8
2.2.	Modelo de fusión JDL	14
2.3.	Niveles de Fusión en Imágenes	15
2.4.	Geometría Proyectiva	19
2.5.	Transformación Proyectiva	20
2.6.	Un ciclo en el estado de Estimación del Filtro Kalman	27
2.7.	Varias medidas, dentro de una región de validación y un solo objetivo válido \hat{Z}^1	29
2.8.	Dos objetivos \hat{Z}^1 y \hat{Z}^2 , con una medida verdadera cada uno y la intersección de sus regiones de validación	30
3.1.	Modelo de Fusión de Imágenes	42
3.2.	Escenario base	42
3.3.	Objetivos nuevos en la escena	42
3.4.	Sustracción de imágenes	43
3.5.	Filtro Sobel para la detección de Bordes	44
3.6.	Umbralización para la detección de las Regiones	44
3.7.	Imagen dividida en Filas y Columnas, tomando en cuenta que Columna es el eje de las $x's$ y Fila es el eje de las $y's$	46
3.8.	Diagrama de Flujo del Algoritmo NNKF	48
3.9.	Diagrama de Flujo del Algoritmo PDA	50
4.1.	Escenario Corredor con los bordes detectados de las regiones	52
4.2.	Escenario Frontal con los bordes detectados de las regiones	52
4.3.	Escenario Corredor con la umbralización. Previamente se realizó una sustracción	52
4.4.	Escenario Frontal con la umbralización. Previamente se realizó una sustracción	52
4.5.	Transformación del escenario Frontal, al plano Corredor	53
4.6.	Puntos localizados dentro del Centro de Proyección, en el escenario Corredor	54
4.7.	Puntos localizados dentro del Centro de Proyección, en el escenario Frontal	54

5.1. Escenario Corredor objeto detectado dentro del centro de proyección . .	70
5.2. Escenario Frontal con el objeto no detectado en el centro de proyección	70
A.1. Modelo de Fusion Ciclo de Inteligencia	79
A.2. Modelo de Fusion Waterfall	80
A.3. Modelo de Fusion Boyd	83
A.4. Interpretación de las entradas y salidas del modelo de Fusión de Datos Dasarathy	84
A.5. Expansión del modelo de Fusión de Datos Dasarathy desde los niveles 0 al 4.	84
A.6. Expansión del modelo de Fusión de Datos Dasarathy desde los niveles 0 al 4.	86
C.1. Formulación matemática del estado de estimación, Estado - Espacio . .	95

Índice de Tablas

3.1. Puntos del escenario Corredor.	45
3.2. Puntos del escenario Frontal.	45
4.1. Escenario Corredor con el objeto detectado	53
4.2. Escenario Frontal con el objeto detectado	53
4.3. Puntos de la función $f(x) = x$	55
4.4. Tabla y gráfica de la ecuación lineal $f(x) = 3x$ y $f(x') = 3x'$	56
4.5. Tabla y gráfica de la ecuación lineal $f(x) = 3x$ y $f(x') = 3x'$ con ruido. En la gráfica siguiente se muestran las diferencias en las funciones originales y estimaciones, con errores. Como se puede apreciar el cálculo de la estimación se vió afectado por el ruido existente generado, sin embargo el algoritmo logró reponerse a los cambios y prosiguió estimando los puntos de manera correcta.	57
4.6. Tabla y gráfica de la secuencia de imagen 1.	58
4.7. Tabla y gráfica de secuencia de imagen 2.	59
4.8. Tabla y gráficas de los puntos asociados del escenario OneLeaveShop1 con el algoritmo NNKF.	60
4.9. Tabla y gráficas de los puntos asociados del escenario OneLeaveShop2 con el algoritmo NNKF.	61
4.10. Tabla y gráficas de los puntos asociados del escenario OneLeaveShop1 con el algoritmo PDA.	62
4.11. Tabla y gráficas de los puntos asociados del escenario OneLeaveShop2 con el algoritmo PDA.	63
4.12. Tabla y gráficas de los puntos asociados a la trayectoria del objeto 1 del escenario EnterExitCrossingPaths1 con el algoritmo JPDA.	65
4.13. Tabla y gráficas de los puntos asociados al objeto 2 del escenario EnterExitCrossingPaths1 con el algoritmo JPDA.	66
4.14. Tabla y gráficas de los puntos asociados a la trayectoria del objeto 1 del escenario OneStopNoEnter2 con el algoritmo JPDA.	67
4.15. Tabla y gráficas de los puntos asociados a la trayectoria del objeto 2 del escenario EnterExitCrossingPaths2 con el algoritmo JPDA.	68
5.1. Puntos de la función $f(x) = x$	71

5.2. Coeficiente de Correlación Lineal en el seguimiento empleando el algoritmo NNKF.	72
5.3. Coeficiente de Correlación Lineal en el seguimiento empleando el algoritmo PDA.	73
5.4. Coeficiente de Correlación Lineal en el seguimiento empleando el algoritmo JPDA.	75

Capítulo 1

Introducción

Cualquier poder, si no se basa en la unión, es débil.

Jean de la Fontaine

La mayoría de las metodologías de investigación y desarrollo de tecnologías se basa en la obtención de muestras, la calidad con que se obtengan depende en gran medida de la fuente que obtiene los datos, esto nos permitirá producir resultados y conclusiones con mayor precisión y exactitud, además de comparar el conocimiento previo, (en caso de que éste exista), con las estadísticas que se hayan generado anteriormente.

Una de las alternativas que permite obtener información de algún escenario a gran escala es la fusión de datos, cuyo objetivo es combinar la mayor cantidad de muestras disponibles provenientes de distintas fuentes de obtención de datos con el menor costo posible.

Regularmente los datos que logran ser obtenidos llegan a ser complementarios de aquellos que no detectaron alguno, esto no genera una estimación estadística de las muestras faltantes, solo es una forma de delimitar la recolección y reconstruir la información que no se tiene. No es un problema de análisis estadístico, pues, solo se consideran los mecanismos que conducen a la ausencia de datos, para la fusión las muestras tienen que ser independientes entre sí [1].

Los sistemas de fusión de datos son actualmente conocidos, aunque la definición del término ha sufrido distintos cambios desde su creación en 1998 Joint Directors of Laboratories, mejor conocido como JDL¹ propuso una definición más general [2].

JDL es uno de los modelos más empleados para categorizar los niveles de fusión de datos cuyo principal objetivo es facilitar la comprensión y la comunicación entre los gestores de adquisición, teóricos, diseñadores, evaluadores y los usuarios de las técnicas de combinación con permiso del diseño de sistemas de costo-efecto, desarrollo y operación [3][4].

¹Joint Directors of Laboratories (JDL) Data Fusion Working Group, fundado en 1985, por el gobierno de EE.UU

Previamente para poder diseñar un sistema de fusión se tienen que tomar en cuenta las limitaciones de cada sensor a utilizar, además de una correcta calibración; si no se verifican estos puntos la fusión podría arrojar información errónea.

Una vez que se han detectado las limitantes en los sensores, el algoritmo a emplear tiene que ser robusto en situaciones donde el escenario no se puede controlar, es decir, la existencia de ruido, las situaciones de oclusión, el contraste y distintos factores atmosféricos que afecten la detección y seguimiento del objetivo en cuestión. Algunos algoritmos solo han sido diseñados para cierto tipo de aplicaciones.

Por lo tanto, un sistema de fusión tiene que resolver tres problemáticas importantes: contar con un conjunto de sensores calibrados, una arquitectura estructurada y un algoritmo de fusión que se adapte a la aplicación [5].

1.1. Planteamiento del Problema

La principal problemática encontrada en lugares donde la afluencia de personas es muy frecuente o zonas aisladas de interés, es la vulnerabilidad de posibles amenazas que estos lugares tienen. Una solución disponible es emplear sistemas de seguridad que se encuentran a cargo de visores humanos resguardando la zona de interés. No obstante las desventajas que estos sistemas presentan son que el personal, a veces, es insuficiente o demasiado costoso para mantener gran cantidad de personal de vigilancia. Cabe mencionar que en estos sistemas se pueden tener dos posibles ambientes: cerrados² y abiertos³.

Una opción eficiente a emplear es la fusión de información de varios sensores, implementada por la arquitectura del modelo JDL, [6], [7], este sugiere que los sensores deben posicionarse de manera distribuida para obtener una topología complementaria, cooperativa, competitiva y redundante para poder realizar una fusión de datos eficiente, en este caso sensores de visión⁴.

Un ejemplo en la utilización de estos sistemas de seguridad esta en los centros comerciales donde la presencia de personas es una constante, por lo tanto, el tener una topología de sensores distribuidos en las zonas de vigilancia facilita el trabajo.

En este caso el modelo JDL se adapta a cualquier aplicación que fusione información, sin embargo, investigadores de visión artificial han creado un modelo de fusión de imágenes el cual se divide en tres niveles claramente definidos[8]:

- *Fusión de datos a nivel Píxel.*
- *Fusión de datos a nivel Características.*

²Aquellos escenarios donde los cambios de luz repentinos o la aparición de sombras estan controlados y el contraste de los individuos u objetos es fácil de distinguir

³Aquellos escenarios donde el ambiente natural de la escena afecta en los cambios de luz y el contraste de fondo se confunde con el del objeto o persona a seguir

⁴Los sensores de visión son cámaras infrarrojas que detectan el calor de los objetos o cámaras filmadoras digitales

■ *Fusión de datos a nivel Decisión.*

Como se mencionó anteriormente los sensores tienen que estar previamente calibrados. Una manera de obtener la calibración es encontrando los puntos de referencia del sensor base para generar una matriz homográfica y poder trasladar los puntos de los otros sensores que están orientados al centro de proyección. La matriz homográfica H , afecta el plano de las imágenes a observar, dependiendo básicamente de cuatro puntos característicos que se tomen en cuenta, para que se hagan corresponder a un solo plano homográfico [9].

Una vez que tiene una arquitectura de los sensores y las cámaras se han calibrado, es necesario emplear un algoritmo de fusión de datos que se adapte a la aplicación, para dar seguimiento a los objetivos que aparecen en cada marco de secuencia.

Varios algoritmos de fusión de datos se han desarrollado y aplicado, de manera individual o combinada, proporcionando a los usuarios distintos niveles de detalle en la información y la fotogrametría ⁵. La elección de un adecuado nivel de la fusión depende del tipo de información disponible, por ejemplo, cuando los sensores son iguales se puede optar por la fusión a nivel píxel, para tomar en cuenta todos los datos. Cuando los sensores o el tipo de información son muy diferentes, la fusión a nivel decisión es más adecuada y también es computacionalmente más eficiente. La fusión a nivel características es viable cuando los datos que se encuentran en el procesamiento provienen de sensores del mismo tipo o son diferentes, pero se pueden asociar adecuadamente [10].

Como se mencionó anteriormente los algoritmos diseñados para aplicaciones de fusión de datos sufren de distintas limitaciones, la aplicación de estos se reduce a cierto tipo de escenarios. El principal objetivo de estos algoritmos es comparar y coleccionar las medidas obtenidas de un objetivo o más. Una de las dificultades es distinguir de que objetivo provienen las medidas, además, si son medidas de objetivos válidos u observaciones generadas por algún fenómeno atmosférico o por el ruido generado por el sensor. Las secuencias de imágenes que se tienen en este caso fueron filmadas en un ambiente cerrado.

Para nuestro caso se sugiere emplear el algoritmo Nearest Neighbor [11], [12], el cual funciona de manera correcta en ambientes controlados. Este algoritmo en imágenes es eficiente donde las variaciones de luz, cambio de contraste y la existencia de ruido son fácilmente identificables y la detección de los objetivos es más fácil; sin embargo, no es tan robusto para darle seguimiento a un objetivo en situaciones no controlables.

Otro algoritmo a emplear y mucho más robusto es el PDA [13], éste asigna una región alrededor de un objetivo ⁶, para hacer un seguimiento de las actualizaciones de

⁵La fotogrametría es la ciencia o técnica cuyo objetivo es el conocimiento de las dimensiones y posición de objetos en el espacio, a través de la medida o medidas realizadas a partir de la intersección de dos o más fotografías, o de una fotografía y el modelo digital del terreno correspondiente al lugar representado, el cual ha de ser realizado anteriormente por intersección de dos o más fotografías.

⁶Esta región es conocida como Gate

las medidas. Una probabilidad de asociación *posteriori* es calculada por cada medida válida. Estas probabilidades se utilizan como ponderaciones, para calcular una mejor media de la actualizada, la cual se agrega a la trayectoria del objetivo. Este algoritmo se desarrolló para la detección y seguimiento de un objetivo.

Ya es sabido que el algoritmo JPDA es una extensión del PDA, este algoritmo es aplicable a sistemas de seguimiento de múltiples objetivos [14]. En este caso las probabilidades de asociación de medidas para el objetivo son calculadas conjuntamente a través de todas las medidas detectadas. Cualquier medida se asume que cae dentro de las regiones de validación, para asegurar que los PDF⁷ de todas las medidas falsas son las mismas o son una nueva muestra.

1.2. Objetivo General

Desarrollar fusión a nivel características, para la detección y seguimiento de personas en un escenario controlado, empleando los algoritmos de asociación NNKF y PDA, para el caso de dar seguimiento de un solo objetivo y el JPDA, para seguir distintos objetivos.

1.3. Objetivos específicos

- *Hacer la detección y seguimiento de objetivos en un ambiente cerrado.*
- *Generar una matriz homográfica que permita una correcta transformación proyectiva para obtener una calibración de las cámaras.*
- *Implementar los algoritmos de asociación de datos NNKF, PDA y JPDA para la combinación de medidas generadas por cada sensor y darle seguimiento a las personas.*

1.4. Organización de la tesis

Los capítulos de esta tesis se han organizado de la siguiente manera:

El Capítulo 2 aborda los fundamentos teóricos implementados en esta tesis, una breve descripción del modelo JDL y el modelo de fusión en imágenes, la segmentación, la generación de la matriz homográfica y la descripción de los algoritmos NNKF, PDA, JPDA.

En el Capítulo 3 Se presenta la propuesta de los algoritmos de segmentación, la generación de la matriz homográfica y los algoritmos de asociación..

⁷Función de Densidad de la Probabilidad, se utiliza en estadística con el propósito de conocer cómo se distribuyen las probabilidades de un suceso o evento, en relación al resultado del suceso. La función de densidad de probabilidad se representa generalmente por $f(x)$.

En el Capítulo 4 se harán pruebas y se discutirán los resultados.
Finalmente en el Capítulo 5 se muestran las conclusiones de esta tesis.

Capítulo 2

Fundamento Teórico

¡Actúa en vez de suplicar. Sacrificate sin esperanza de gloria ni recompensa! Si quieres conocer los milagros, hazlos tú antes. Sólo así podrá cumplirse tu peculiar destino. Ludwig van Beethoven

2.1. Fusión de Datos

Anteriormente se habían utilizado definiciones para el término Fusión de Datos, pero estas eran exclusivas para ciertas áreas de investigación, por ejemplo en [15] y [16], se definió solo para el caso de fusión de imágenes y en [17] [18], [19], [20], [21] se hace una revisión muy detallada sobre la definición de fusión de datos multi - sensor. En 1998, Joint Directors of Laboratories dió a conocer una definición general de fusión de datos.

La fusión de datos es un marco formal en el que se expresan los medios y las herramientas para la unión de los datos procedentes de diferentes fuentes. Su objetivo es la obtención de información de mayor calidad, la definición exacta de “mayor calidad” dependerá de la aplicación...[22]

Actualmente en internet se puede encontrar una definición similar a la mencionada anteriormente[23], no hay gran diferencia en este término y el dado por JDL.

2.1.1. Aplicaciones de los modelos multisensor

La fusión multisensor es un término utilizado para los sistemas de fusión que emplean sensores del mismo tipo o sensores diferentes [24], [25], [26]. Esta área de investigación tiene tres campos relevantes de aplicación: i) Identificación de objetos o personas, ii) Seguimiento de objetos o personas, iii) Aplicaciones en robótica.

Las primeras aplicaciones de los sistemas de fusión eran militares y el principal objetivo era la detección y seguimiento de enemigos así como la vigilancia de ciertas zonas de acceso restringido [27], [28], [29], [30], [31] [32]; sin embargo los ejemplos en

el área civil son numerosos, por ejemplo, en el transporte, la fusión se ha visto aplicada en el control del tráfico aéreo y vigilancia [33], [34], [35].

Actualmente los sistemas más sobresalientes de la fusión de datos se pueden observar en los sistemas de navegación - posicionamiento, es decir, la navegación por GPS [36],[37] o los sistemas empleados en los autos inteligentes [38], el cual es un trabajo en conjunto de algún satélite en órbita y los dispositivos en Tierra. Los teléfonos, en el caso de los medios de comunicación a través de complejos sistemas de fusión para realizar una llamada telefónica [39] y en muchas otras aplicaciones que emplean distintos dispositivos para fusionar información [40], [41], [42].

Por mencionar un ejemplo, en las actividades de cartografía, a menudo se utilizan algunas características donde se tiene lugar, en este caso, un sistema de información geográfica que ayuda en la clasificación de imágenes multispectrales generadas por varios sensores a bordo de vehículos espaciales o aéreos [43], [44]. En esta situación particular, algunos datos son mediciones de la energía electromagnética y otros pueden ser solo simbólicos, como en el caso de la Arqueología [45].

La robótica es otro ejemplo del desarrollo de los sistemas de fusión, [46] [47], por ejemplo, los datos recibidos de un escenario generado en 3D, son eficientes cuando el robot se desplaza en un ambiente hostil, para la inspección y mantenimiento de piezas o artefactos peligrosos, donde el hombre no puede actuar, el sistema robótico puede solucionarlo o en el caso de la medicina [48], donde con base en imágenes el algoritmo puede detectar síntomas de algunas enfermedades o los robots pueden realizar una operación de riesgo.

Otra aplicación se puede dar en el desarrollo de tecnologías para analizar y combinar las muestras biométricas de las personas [49], [50].

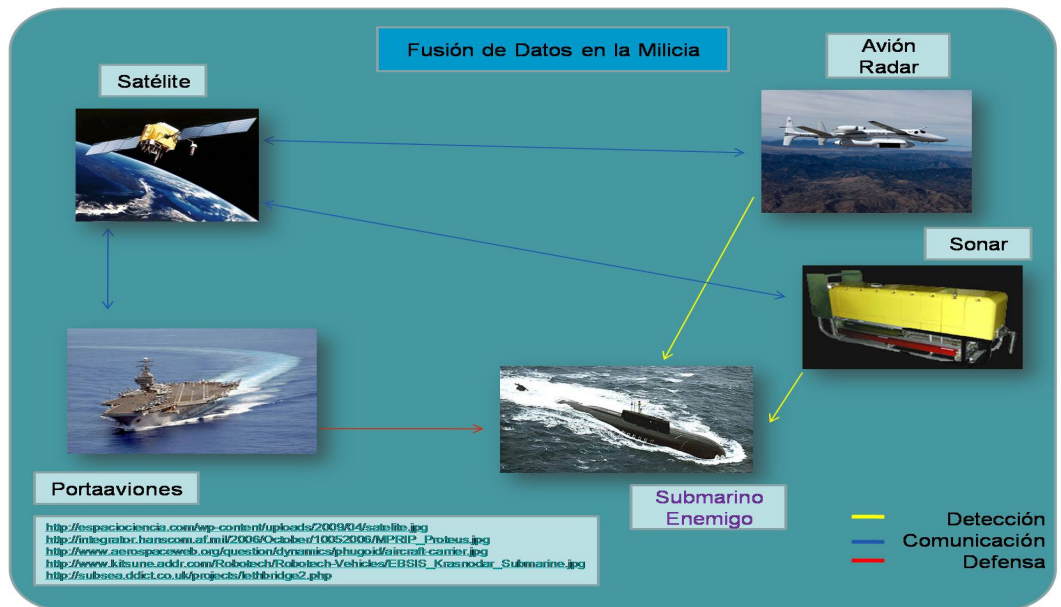


Figura 2.1: Fusión de distintos tipos de sensores

El pronóstico del tiempo ilustra plenamente la fusión de datos en el medio ambiente [51], [52]. Es uno de los más sofisticados sistemas de fusión. Se exhorta a los sensores, procesamiento de señales, inteligencia artificial y complejos modelos de la Física, Química, la Atmósfera, los Océanos y la Tierra.

Las exploraciones realizadas a la Tierra, por medio de satélites, aviones, buques, etc. hacen que la fusión se realice de manera redundante y complementaria. En general la disponibilidad de tantas fuentes de información, (radares, globos, aviones, boyas automáticas, sensores de temperatura, presión, medios especializados de comunicación, modelos numéricos, etc) constituyen un enorme campo de investigación para los desarrolladores de modelos de fusión. Este interés se ve reforzado por la forma correcta del modelado de los paisajes naturales y escenas al aire libre.

2.1.2. Ventajas de sistemas multisensores

Cabe mencionar que la fusión de datos se puede obtener de dispositivos que cumplan con las tres características de un buen sistema de fusión mostrado en [53], [54], [55], [56].

- *Sensores calibrados.*
- *Arquitectura ó Topología de los sensores.*
- *Algoritmo adecuado a la aplicación.*

Las ventajas obtenidas de la fusión de datos con respecto a la utilización de un solo sensor, se listan a continuación.

- *Si varios sensores idénticos se utilizan (por ejemplo, los radares de seguimiento de un objeto en movimiento), la combinación de las observaciones se traducirá en una mejor estimación de la meta la posición y velocidad. Una ventaja estadística se ha obtenido mediante la adición de N observaciones independientes (por ejemplo la estimación de la ubicación de destino o la mejora de velocidad es por un factor proporcional a N), suponiendo que los datos se combinan de manera óptima. Este resultado también puede ser obtenido mediante la combinación de N observaciones de sensores diferentes.*
- *Una segunda ventaja implica el tipo de arquitectura empleada o una combinación de múltiples sensores para mejorar el proceso de observación. Por ejemplo, dos sensores que miden las direcciones angulares de un objeto pueden ser coordinados a fin de determinar la posición de un objeto por la triangulación. Esta técnica se utiliza en el estudio y navegación comercial. Del mismo modo, la utilización de dos sensores, uno en movimiento en una forma conocida con respecto a otro, se puede utilizar para medir instantáneamente la posición de un objeto y la velocidad con respecto a la observación de los sensores.*

- *Una tercera ventaja adquirida es mediante el uso de múltiples sensores que mejoran la observabilidad. La ampliación de la línea base de observaciones físicas puede dar lugar a mejoras significativas. Por ejemplo, un objeto en movimiento, como un avión, que se detecta mediante los impulsos de un radar y una visión de un sensor de imagen infrarroja (FLIR). El radar puede determinar con precisión la distancia, altitudes, dirección y velocidad, pero tiene una capacidad limitada para determinar la dirección angular de la aeronave. Por el contrario, el sensor de imágenes infrarrojas puede determinar con precisión la dirección angular de la nave. Si estas dos observaciones se asocian correctamente, la combinación de los dos sensores proporciona una mejor determinación de la ubicación que podría obtenerse por cualquiera de los dos sensores de manera independiente. Esto se traduce en una reducción de error de región, como se muestra en la estimación fusionada o combinada. Un efecto similar puede ser obtenido en la determinación de la identidad de un objeto basado en observaciones de los atributos del objeto. Por ejemplo, hay pruebas de que los murciélagos identifican a sus presas mediante una combinación de factores, incluyendo el tamaño, textura (basado en la firma acústica), y el comportamiento cinemático.*

Con estos puntos mencionados es fácil detectar que las mediciones en un modelo de fusión de datos se pueden realizar de manera puntual y en tiempo integral, la extracción de información puede ser de manera bidimensional e instantánea (imágenes), se pueden generar perfiles verticales con el tiempo de integración o no, en tres dimensiones (océánico, perfil de la atmósfera, sólida a nivel del suelo, por satélite o algún buque a carga, etc). La generación de la estadística y los modelos matemáticos representan para los científicos mayor cantidad de información disponible para describir el resultado.

Los problemas posiblemente encontrados en los modelos de fusión se deben al nivel de procesamiento de la información, problemas con la arquitectura (la distribución de los sensores en el espacio 3D y el desfase de tiempo) y problemas con la transferencia de datos [57].

2.2. Modelo de Fusión de Datos

La fusión de datos es un campo de investigación que emplea diversos métodos de investigación operacional, ingeniería industrial, matemáticas computacionales, probabilidad, estadística, comunicación y teoría de decisión, lógica difusa, gestión de incertidumbre, teoría de estimación, procesamiento de imágenes digitales, procesamiento de señales digitales y las funciones de la Inteligencia Artificial, etc. por lo tanto en esta sección se dará una breve revisión a uno de los modelos de fusión más utilizado en los últimos años así como una descripción.

El modelo Joint Directors of Laboratories Data Fusion Working Group, mejor conocido como JDL, fundado en 1985, por el gobierno de EEUU [3], [58], [59]. fue creado con la intención de categorizar las distintas etapas de un proceso de fusión de datos,

las arquitecturas generadas con este modelo son: independiente, redundante, complementaria y cooperativa. Por ejemplo, un sistema de fusión diseñado por un conjunto de sensores de distinto tipo, cada uno de ellos proporciona información de manera independiente y diferente. La integración o combinación de datos daría como resultado información complementaria y cooperativa. Empleando múltiples sistemas similares a este, se obtienen arquitecturas redundantes.

Como se mencionó las muestras obtenidas por las distintas fuentes son independientes, por lo tanto, cada estimación generada por cada una de estas fuentes es complemento de alguna otra. El objetivo primordial, es generar una sola estimación del resultado.

2.2.1. Modelo JDL (Joint Directors of Laboratories):

Este modelo¹ ilustra las principales funciones, la información pertinente, las bases de datos y la interconexión existente para llevar a cabo la fusión de datos [60], [61], [62], [63], [64], [65], [66], [67], [68]. Se compone de 6 niveles, listados a continuación:

- *Nivel 0 Pre- Proceso o Proceso de Asignación: La tarea de este nivel es reducir la carga de procesamiento de fusión en los procesos adecuados de preselección y asignación de señales. Las técnicas utilizadas en este nivel pertenecen a la transformación de la señal y extracción de características. En este nivel, ningún significado semántico se asigna a los datos evaluados.*
- *Nivel 1 Objeto de Evaluación ó Refinamiento del objeto²: En este nivel se desarrolla la estimación y predicción del estado de entidad sobre la base de observación para la asociación de la trayectoria del objetivo, estado continuo de estimación y estado discreto de estimación, (ejemplo: procesamiento de datos).*

Esta conformado por los siguientes subprocesos:

- *Alineación de Datos: Alinea los datos recibidos de múltiples sensores a un marco de referencia común. La alineación facilita la comparación y el procesamiento de múltiples alertas que provienen de diferentes fuentes y es muy probable que posean diferentes características. El marco de referencia, por lo tanto, debe proporcionar suficiente espacio para los diferentes tipos de características para ser representados. La fase de adaptación requiere la siguiente información.*
 - *1. Alertas crudas, es decir, datos en bruto que pasaron en el Nivel 0 de transformación.*
 - *2. Un marco de referencia común para que las descripciones puedan ser coherentes;*

¹Fue desarrollado en 1985 por los EE.UU. en el Joint Directors of Laboratories (JDL) Data Fusion Group

²Es denominado a veces alineación de los datos.

- 3. Un procedimiento de alineación.
 - *Asociación: Combina, ordena o correlaciona múltiples observaciones de sensores que se relacionan a una sola entidad. La dificultad esta en distinguir el objeto real de la trayectoria. Es decir, en un sistema distribuido de asociación se puede tener el paso de varias trayectorias donde los nodos de procesamiento pueden ser comparados, para poder combinar la trayectoria del estado del objetivo. Es una asociación de seguimiento de una trayectoria.*
 - *Rastreo ó Seguimiento: Consiste en la combinación de múltiples observaciones de la posición de los datos para estimar la posición y la velocidad de una entidad.*
 - *Identificación: Combina los datos relacionados con la identidad para refinar la estimación de las identidades de una entidad o la clasificación. Es decir que de acuerdo a las mediciones obtenidas clasifica al objeto. Los datos son obtenidos de los sensores para obtener una mejor estimación del objetivo identidad, aunado a un valor de confianza para la identidad. Una vez estimado en los sensores locales, se debe estimar con sensores remotos, para obtener una solución global.*
- *Nivel 2 Situación de Refinamiento ó Evaluación: En este nivel se lleva a cabo la estimación y predicción de las relaciones entre los estados de un objetivo a fin de incluir las bases que infieren las relaciones de las entidades, es decir, intenta encontrar la descripción contextual de las relaciones entre los objetos y los eventos observados, (ejemplo: Procesamiento de la información).*
- *Nivel 3 Evaluación del Impacto ó Procesamiento: Basado en el conocimiento previo y las predicciones sobre las situaciones futuras este nivel de procesamiento intenta obtener las inferencias sobre las vulnerabilidades y oportunidades por operación. Estima el impacto de la situación evaluada, es decir el impacto de diversos planes ya que interactúan entre si y con el entorno. Este Nivel es considerado un subconjunto del Nivel 2, considerando que el Nivel 2 implica estimar o predecir todos los tipos de estados relacionales, el Nivel 3 involucra la predicción de algunas de las relaciones entre un usuario específico y su entorno, incluyendo la interacción con las acciones de otros usuarios, dado el plan de acción del usuario y para cualquier otro usuario.*
- *Nivel 4 Proceso de Refinamiento: Es un metaproceso que tiene como objetivo optimizar el rendimiento general del sistema de fusión. Se compone de cuatro pasos clave:*
 - *Evaluación del desempeño: Proporciona información sobre el control en tiempo real y rendimiento a largo plazo.*

- *Control de Procesos: Identifica la información necesaria para mejorar el producto de la fusión de múltiples niveles.*
- *Determinación de la Fuente de Requerimientos: Determina la fuente de los requisitos específicos para obtener la información pertinente.*
- *Gestión de la Misión: Asigna y dirige las fuentes para lograr los objetivos de la fusión.*

Este Nivel a veces se denomina Proceso de Adaptación para enfatizar que es dinámico y debe ser capaz de evolucionar con respecto a sus propiedades internas y el medio ambiente circundante. Interactúa con cada uno de los demás niveles ya que implica la asignación de recursos a las tareas.

- *Nivel 5 - Refinamiento Cognitivo: Se dice que es un elemento del gestor de conocimiento, determina quien requiere la información y quien tiene acceso a ella, además recupera los datos de adaptación y muestra el soporte cognitivo en la toma de decisiones y acciones. Este proceso involucra un número de subprocesos y atributos, incluyendo representaciones internas y su manipulación, atención, memoria, formación de conceptos, conocimiento, generalización, reconocimiento, entendimiento, significado, predicción, imaginación, intuición, emociones, decisiones, razonamiento, metas, comportamientos, conciencia, inconciencia, etc. un subconjunto mínimo de estos procesos que involucra mecanismos para las señales entrantes, (señal externa del sensor) y salientes (representación interna de la señal). En sí cualquier reconocimiento y proceso de formación del concepto involucra un razonamiento entre estos dos tipos de señales.*

Existen distintos modelos de fusión: los centralizados, distribuidos e híbridos [69], [70], la mayoría se divide en niveles de fusión y tiene similitudes; en el Apéndice A, se pueden encontrar descripciones breves de algunos de estos Modelos de Fusión.

2.3. Fusión en Imágenes

En esta sección se define la Fusión de Imágenes y los niveles en los cuales se divide.

La Fusión de imágenes se define como el proceso de combinar la información complementaria de múltiples imágenes fuente, para obtener una imagen resultante. El algoritmo que se emplee en estas imágenes, para que se combinen, depende en gran parte a la forma en que estas fueron obtenidas, [71], [72], [73], [74], [75].

Este proceso esta dividido en tres categorías, debido al tipo de procesamiento al que pueden ser sometidas las imágenes, [76], [77], [78], [79].

- *Fusión de datos a nivel Píxel: Es la presentación de varias bandas espectrales de una misma escena, las cuales se combinan para formar una nueva imagen, que idealmente, contiene un solo canal con toda la información disponible de*

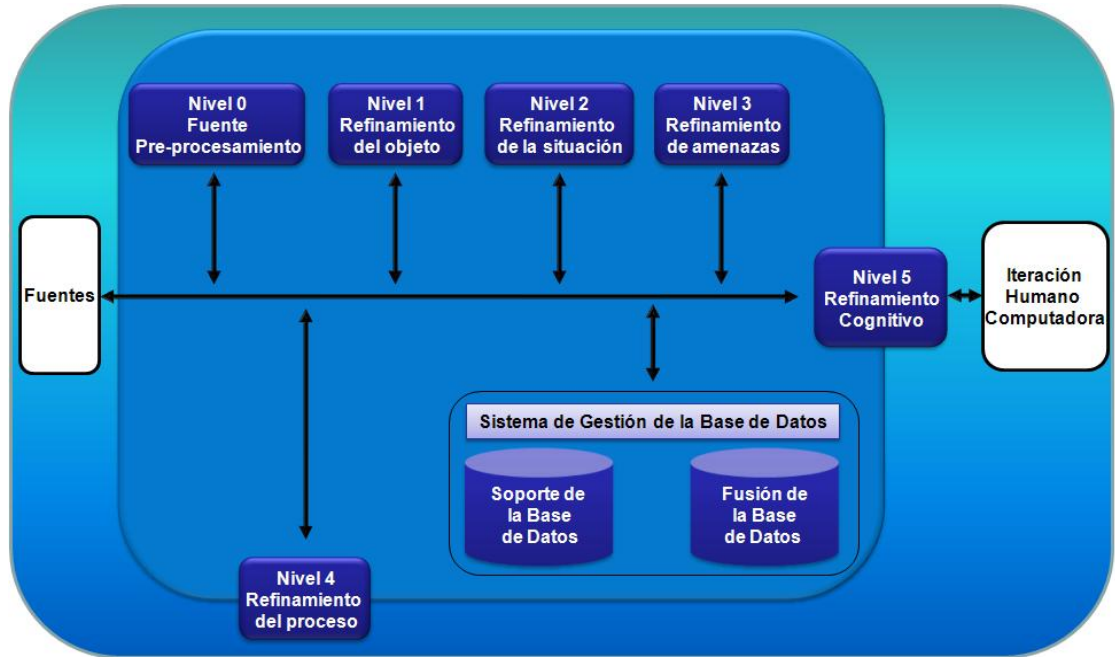


Figura 2.2: Modelo de fusión JDL

las distintas bandas espectrales. Al final el proceso podría utilizar esta imagen única en lugar de las imágenes originales. Es particularmente importante cuando el número de bandas espectrales disponibles se hace tan grande que es imposible ver las imágenes por separado. Y más si estas imágenes provienen de distintos tipos de sensores, ya sea, infrarrojo o visible.

Ventajas:

- Existe correlación entre las fuentes empleadas, fusiona la información pixel por pixel.
- Es adecuado para la detección de cambios.

Limitaciones:

- Asume que los datos pueden ser modelados a una función de densidad de probabilidad común.
 - No es adecuado para datos heterogéneos.
- *Fusión a nivel características:* Se le llama así debido a que fusiona varias características, las cuales pueden provenir de varias fuentes de datos o de los propios datos. En el segundo caso, el objetivo es encontrar las características disponibles que podrían venir de varios métodos de extracción. La tarea en sí, es obtener un número limitado de características relevantes. Normalmente en el procesamiento de imágenes, los mapas de características se calculan previos a la detección.

Estos mapas se calculan y se combinan para que luego puedan ser utilizados en la detección. Algunas investigaciones han demostrado que las técnicas de fusión que operan con fusión a nivel características han obtenido mejores resultados que aquellas generadas a nivel píxel.

Ventajas:

- El sensor de características específicas da ventaja sobre la fusión basada en píxeles.
- Muy adecuado para los cambios en la detección.

Limitaciones:

- Asume que los datos pueden ser modelados por una función de densidad de probabilidad común.

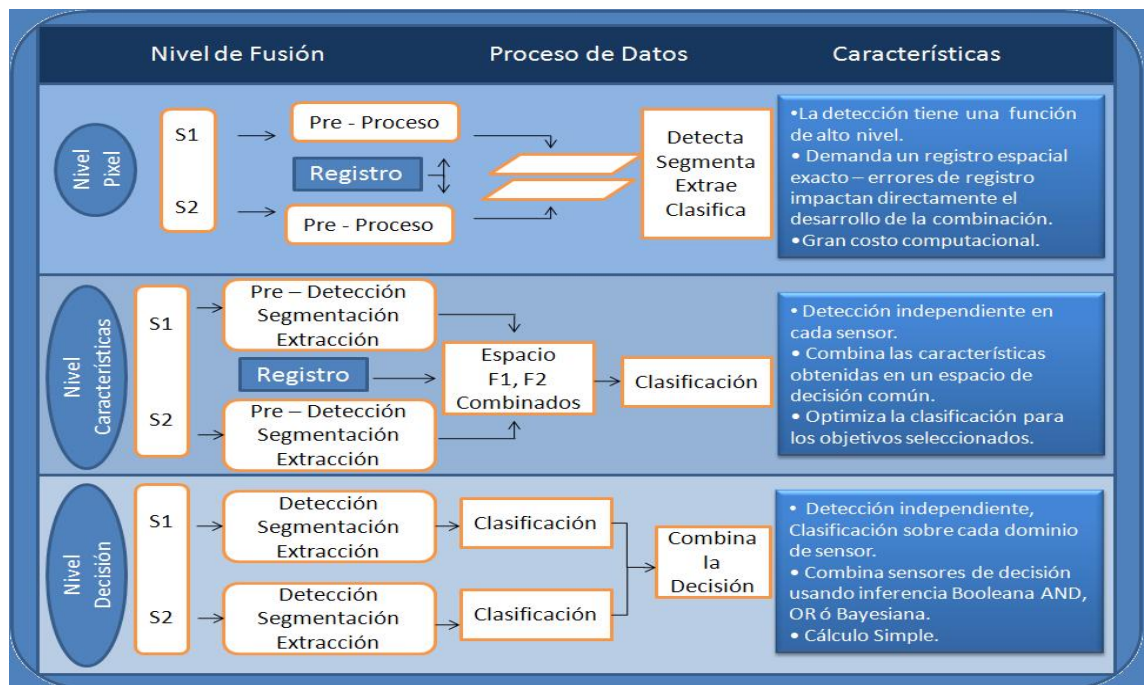


Figura 2.3: Niveles de Fusión en Imágenes

- **Fusión a nivel Decisión:** Combina las decisiones que provienen de distintas fuentes de información. Por extensión, uno habla de fusión a nivel decisión, si el “experto” regresa un estimado del evento y no una decisión. Para poder distinguir ambos métodos de decisión se clasifican en: i) Toma de decisiones difíciles, (simple elección óptima) y ii) las decisiones suaves, en donde las decisiones de incertidumbre en cada sensor se mantienen y combinan con una medida compuesta por la incertidumbre. El comportamiento relativo de reglas de combinación alternativas

y los umbrales del sensor independiente pueden ser seleccionados de forma óptima a partir de datos de distribución de las características utilizadas por cada sensor. En la fusión a nivel decisión, cada sensor debe detectar independientemente la presencia de un objetivo candidato y desarrollar una clasificación sobre este candidato. Estas detecciones y/o clasificaciones, (del sensor de decisión), son combinadas dentro de la decisión fusionada.

Ventajas:

- *Adecuado para datos con diferentes densidades de probabilidad.*

Limitaciones:

- *Clasifica cada imagen independientemente a veces esto consume más tiempo, para luego hacer la combinación preliminar.*

2.4. Segmentación

La segmentación se refiere al proceso de agrupar una imagen digital en múltiples regiones. La meta es simplificar y/o cambiar la representación de una imagen dentro de algo más significativo y fácil de analizar. La segmentación es empleada para localizar objetos, en la estimación de los límites de oclusión dentro de una imagen, en la comprensión de imágenes, en la edición de imágenes, etc. hay dos alternativas básicas, encontrar los bordes o determinar las regiones, [80], [81], [82], [83].

La segmentación es un módulo necesario del sistema de procesamiento de imágenes, pues toma medidas sobre las regiones detectadas ya sea para reducir la información de la imagen o para clasificar y hacer un diagnóstico automático de la información que se pueda generar a partir de la detección.

2.4.1. Métodos de Segmentación

En la segmentación de imágenes se pueden encontrar distintas técnicas que han sido desarrolladas debido al tipo de proceso con que se quiera procesar la imagen o a la naturaleza de esta, las técnicas mencionadas a continuación segmentan objetos o regiones con imágenes a niveles de gris, aunque algunos pueden ser utilizados para segmentar imágenes a color [84], [85], [86]:

- Métodos de Segmentación de Bordes
- Métodos Basados en Histograma ó Umbral
- Métodos Basados en Árbol / Grago

- Métodos de División de Regiones
- Métodos de Crecimiento de Regiones
- Segmentación Basada en Modelo
- Segmentación Basada en Redes Neuronales
- Métodos de Agrupamiento
- Métodos de Partición de Grafos
- Por la Transformación de Watershed
- Segmentación Multi - escala
- Aproximaciones Bayesianas y Probabilísticas.

2.4.2. Problemas de la segmentación

Para el ser humano le es más fácil detectar con la vista algunos de los objetos en los distintos escenarios, excepto en aquellos con escasa luz o exceso de esta. En el procesamiento de imágenes por computadora, la detección de objetos no es un proceso trivial, algunos problemas son generados por distintos factores, algunos de ellos están listados a continuación:

- Oclusión: Se presenta cuando uno o más objetos obstruyen la visibilidad del objetivo que se está siguiendo, por lo tanto llega a distorsionar la información.
- Volumen Parcial: Las imágenes al ser representadas en una dimensión en 2D de objetos en 3D, la información encontrada en la imagen no pertenece propiamente al plano, este problema limita la definición de las regiones.
- Falta de homogeneidad: Los objetos no poseen siempre una iluminación uniforme, a veces es una característica de textura que los diferencia de otros.
- Forma: La forma de los objetos varía demasiado, tanto pueden tener una forma regular y convexa, o ser objetos finos y muy ramificados o totalmente irregulares, con bordes no definidos.
- Ruido: La imagen puede ser de baja calidad debido a la presencia de ruido o a una mala adquisición de la misma.

Los problemas de ambigüedad encontrados en la imagen pueden afectar la segmentación de la misma, por lo tanto se tiene que tomar en cuenta el preprocesamiento para mejorar la calidad de la imagen, con el objetivo de obtener información confiable y la información encontrada en ella sea únicamente de la imagen.

2.5. Geometría Proyectiva

Para ver por qué se va a emplear la geometría proyectiva se parte desde la geometría euclidiana familiar. Esta es la geometría que describe los ángulos y las formas de los objetos. En la Geometría Euclidiana se sigue haciendo una excepción a la razón sobre algunos de los conceptos básicos de la geometría, como la intersección de las líneas. Dos líneas, (se piensa en geometría bidimensional), casi siempre se encuentran en un punto, pero hay algunos pares de líneas que no lo hacen, las cuales se les conoce como paralelos. Las líneas paralelas se llegan a tocar en “el infinito”. Sin embargo, esto no es totalmente convincente. Se puede evitar esto mediante la mejora del plano euclidiano mediante la adición de estos puntos en el infinito, donde las líneas paralelas se encuentran y resuelven las dificultades con el infinito, llamándose “puntos ideales”. Al añadir estos puntos en el infinito, el Espacio Euclidiano familiar se transforma en un nuevo tipo de objeto geométrico, el Espacio Proyectivo. El espacio proyectivo, es una extensión del espacio euclidiano en el que dos líneas siempre se unen en un punto, aunque a veces estos puntos se encuentren en el infinito [87].

Coordenadas: Un punto en el espacio euclidiano 2D, está representado por un par ordenado de números reales, (x, y) . Se puede añadir una coordenada extra a ésta pareja, dando $(x, y, 1)$, que declara representar el mismo punto. Se puede ir adelante y atrás de una representación del punto a otro, simplemente añadiendo o eliminando la última coordenada. Es aquí donde se hace una definición y se dice que $(x, y, 1)$ y $(2x, 2y, 2)$ representan el mismo punto, y además, (kx, ky, k) representa el mismo punto, para cualquier valor distinto de cero de k . Formalmente, los puntos están representados por clases de equivalencia de coordenadas triples, donde dos triples son equivalentes cuando se diferencian por un múltiplo común. Estos se llaman las coordenadas homogéneas del punto. Dada una coordenada triple (kx, ky, k) , podemos obtener las coordenadas originales de nuevo dividiendo por k para obtener (x, y) .

Homogeneidad: En la geometría euclidiana clásica todos los puntos son los mismos. La totalidad del espacio es homogéneo. Cuando se agregan coordenadas, aparentemente un punto es elegido como el origen. Considerándose un cambio de coordenadas para el Espacio Euclidiano en donde los ejes se desplazan y se giran a una posición diferente. Esto se traduce que en el espacio como la traslación y la rotación. La operación resultante se conoce como una Transformación Euclidiana. Un tipo más general de transformación es la aplicación de una transformación lineal \mathbb{R}^n , seguida de una transformación euclidiana moviendo el origen del espacio. Esto se tiene como el espacio de movimiento, la rotación y, finalmente, se extiende de forma lineal, posiblemente, por diferentes razones en diferentes direcciones. La transformación resultante se conoce como una transformación afín. El resultado ya sea de un euclidiana o una transformación afín es que los puntos en el infinito permanecen en el infinito. Estos puntos son de alguna manera, un conjunto de estas transformaciones. Estos son distinguibles o especiales en el contexto de la geometría euclidiana o afín, véase apéndice B.

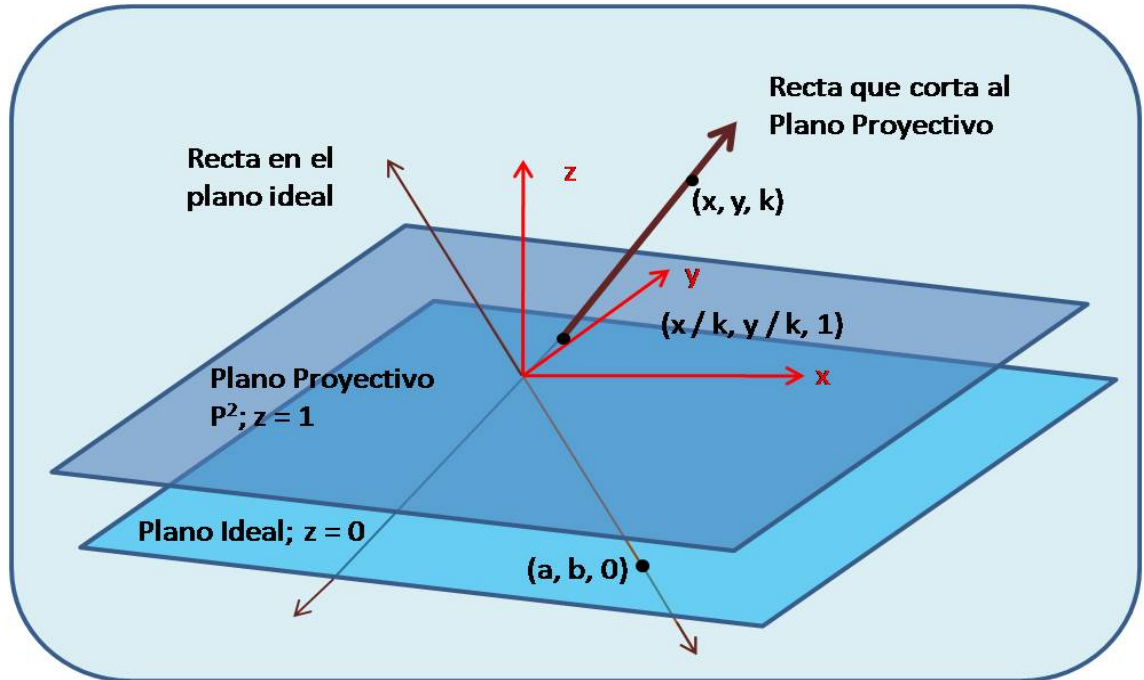


Figura 2.4: Geometría Projectiva

2.5.1. Transformación Projectiva

Desde el punto de vista de la geometría projectiva, los puntos en el infinito, no son diferentes de otros puntos. Así como el espacio euclidiano es uniforme, lo es el espacio projectivo. Los puntos en el infinito, tienen como última coordenada cero, en sí es una representación homogénea de coordenadas. Por analogía con las transformaciones euclidianas o afines, se puede definir una transformación projectiva del espacio projectivo. Una transformación lineal del espacio euclidiano \mathbb{R}^n está representado por la multiplicación de matrices aplicados a las coordenadas del punto. En la misma manera que una transformación projectiva del espacio projectivo \mathbb{R}^n es una aplicación homogénea de las coordenadas de un punto $(n + 1) - vector$, en la que el vector de coordenadas se multiplica por una matriz no singular. Bajo este tipo de cartografía, los puntos en el infinito (con la coordenada final cero) se asignan a otros puntos arbitrarios. Los puntos en el infinito no se conservan. Por lo tanto, una Transformación Projectiva del Espacio Projectivo \mathbb{P}^n está representada por una transformación lineal de las coordenadas homogéneas,

$$X' = H_{(n+1) \times (n+1)} X$$

La consecuencia básica de esta linealidad son mapas hiperplanos de Homografías para los hiperplanos. Generalmente estos mapean cualquier subespacio projectivo para un subespacio de la misma dimensión, una propiedad la cual es llamada conservación de incidencia. Las Homografías forman un grupo llamado $\mathcal{P}\mathcal{L}\mathcal{G}_n$, el cual es nombrado Grupo Projectivo. Dadas dos bases projectivas hay una única homografía que mapea

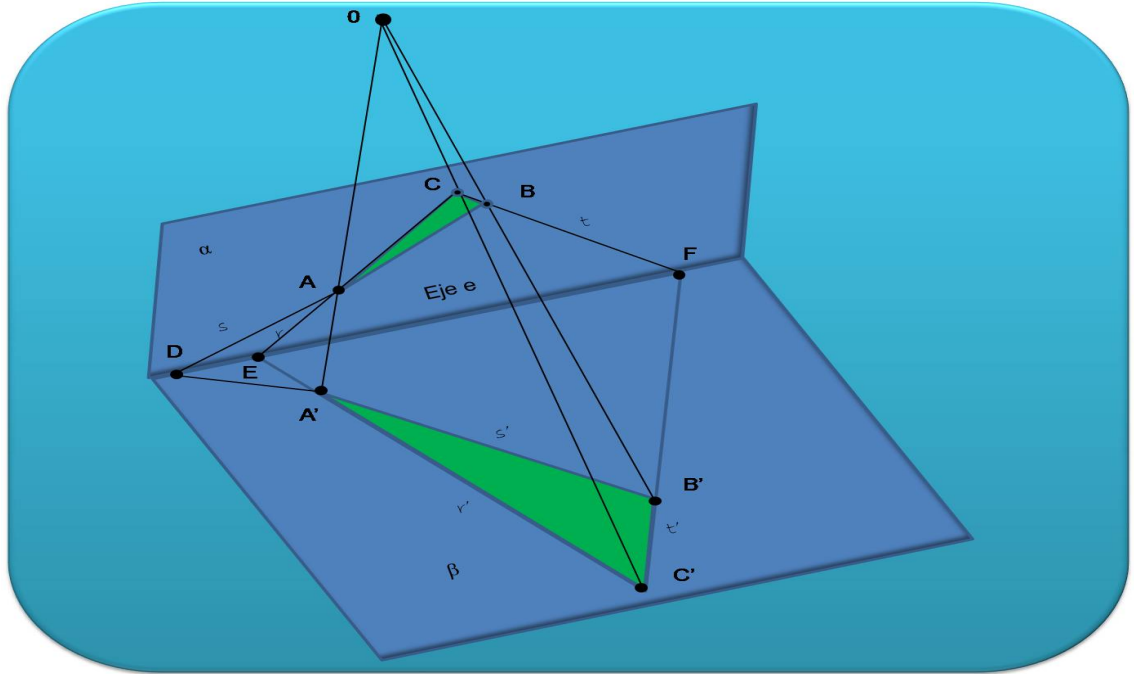


Figura 2.5: Transformación Projectiva

los puntos de la primera base a los de la segunda, por lo tanto las homografías se pueden considerar como las transformaciones que cambian el desempeño de las coordenadas proyectivas [88], [89].

Se observará que si bien $(x, y, 1)$ representa el mismo punto que el par de coordenadas (x, y) , no tiene sentido hacerlo corresponder con $(x, y, 0)$. Si tratamos de dividir la última coordenada, se tiene el punto $(x/0, y/0)$, que es infinita. Así es como los puntos en el infinito se plantean a continuación. Son los puntos representados por las coordenadas homogéneas en las que la última coordenada es cero. Una vez que se ha visto cómo hacer esto para el espacio euclidiano 2D, se extiende a un espacio proyectivo mediante la representación de los puntos como vectores homogéneos, es evidente que no se puede hacer lo mismo en cualquier dimensión. En pocas palabras el espacio euclidiano \mathbb{R}^n puede extenderse a un espacio proyectivo \mathbb{P}^n mediante la representación de los puntos como vectores homogéneos. Resulta que los puntos del infinito en las dos dimensiones forman el espacio proyectivo de una línea, generalmente se llama la línea en el infinito.

2.5.2. Determinación de la Matriz Proyectiva Homogénea 2D

La matriz debe de cumplir de forma óptima las correspondencias de al menos 4 puntos.

$$H = \begin{bmatrix} P_x \\ P_y \\ P_w \end{bmatrix} \begin{bmatrix} Q_x \\ Q_y \\ Q_w \end{bmatrix} \quad (2.1)$$

Si menos de 4 pares de puntos $(P_x, P_y, P_w), (Q_x, Q_y, Q_w)$ se dan, no existe una solución única, si los pares son exactamente 4 se obtiene la matriz de transformación de la forma deseada, y si hay más de 4 pares de puntos dados, se busca minimizar el error de transformación.

Si los puntos finitos se utilizan, por lo general P_w y Q_w se establecen en 1. H tiene la ventaja de ser un método de optimización adicional que se puede utilizar y las covarianzas de los puntos pueden ser tenidos en cuenta.

Si los puntos que transforman la imagen se especifican en el estándar de las coordenadas, las coordenadas de fila deben pasar de P_x y las coordenadas en la columna P_y . Esto es necesario para obtener coordenadas de forma correcta con la imagen. En particular, esto asegura que las rotaciones se realizan en la dirección correcta. Se debe de tomar en cuenta que las coordenadas (x, y) ordenan las matrices de manera natural y se hace corresponder con el habitual (fila, columna) para las coordenadas de la imagen.

Cabe señalar que las matrices de transformación homogénea se refieren a un general de las que se encuentran amano derecha de la ecuación es decir, de coordenadas matemáticas del sistema. Si una matriz homogénea de transformación se utiliza para transformar las imágenes, las regiones, contornos XLD, o cualquier otro dato que ha sido extraída de las imágenes, la fila de las coordenadas de la transformación debe ser aprobada en las coordenadas x , mientras que la columna de las coordenadas debe ser aprobada en las coordenadas y . En consecuencia, la orden de pasar las coordenadas de fila y columna sigue el orden habitual (Fila, Columna). Esta convención es esencial para obtener coordenadas del sistema para la transformación de los datos icónicos, y en consecuencia para garantizar, en particular, que las rotaciones se realizan en el sentido matemático correcto.

2.6. Estimación, Decisión y Seguimiento con Múltiples Sensores

La tarea fundamental de los algoritmos de predicción y asociación de datos para el seguimiento de uno o más objetivos es asumir la probabilidad de detección de los objetivos, ya sean conocidos o desconocidos, usando una configuración centralizada, es decir, el algoritmo recibe toda la información central para procesarla, para poder llegar a este punto se tienen que tomar en cuenta algunos procesos inmersos encontrados en

las distintas etapas de la formación de estos objetivos, esto se resume de la siguiente manera ³ [90].

- **Detección Inicial.**
- **Adquisición del Objetivo - Formación del seguimiento ó iniciación.**
- **Precisión del Seguimiento - Mantenimiento del seguimiento ó continuación.**

Una vez realizados estos puntos se encuentran distintas problemáticas a resolver clasificados de acuerdo al tipo de criterio:

- **Etapas de Rastreo ó Seguimiento:**
 - Formación del rastreo, (iniciación).
 - Mantenimiento del rastreo, (continuación).
- **Número de Sensores:**
 - Solo un sensor
 - Varios sensores.
- **Características del detección del sensor:**
 - Probabilidad de detección del objetivo, (P_D).
 - Probabilidad de falsas alarmas, (P_{FA}).
- **Comportamiento del Objetivo:**
 - Estable, (maniobrable).
 - Inestable, (inmaniobrable).
- **Número de objetivos:**
 - Solo un objetivo.
 - Múltiples objetivos.
- **Tamaño del objetivo:**
 - Depende de la resolución del sensor.
 - Objetivo de gran extensión.
 - Tipo de imagen.

En esta sección se describirán algunas situaciones comunes de los algoritmos de seguimiento empleando varios sensores, ya que se evalúa la asociación de medidas para rastreos por la vía de regiones de validación. Obteniendo así el seguimiento de solo un objetivo o varios.

³La información mostrada en esta sección fué tomada del libro **Multitarget-Multisensor Tracking: Principles and Techniques**, de Yaakov Bar-Shalom y Xialo-Rong Li

2.6.1. Seguimiento con múltiples sensores

Cuando se obtienen medidas de múltiples sensores se tienen que distinguir diferentes casos a resolver, entre ellos se mencionan los siguientes:

- Sensores sincronizados, los tiempos de muestreo de los sensores son los mismos.
- Sensores asíncronos, los tiempos de muestreo son arbitrarios, (casos reales).

Si los sensores están sincronizados y el ruido en las medidas proveniente de los sensores no está correlacionado (en un sistema real), entonces los dos esquemas se actualizan de dos formas.

- Secuencial: Con los datos desde un sensor para un mismo tiempo.
- Paralela: Simultáneamente con las medidas acumuladas.

Ambos son equivalentes para los **sistemas lineales**, donde la actualización secuencial es menos cara que la paralela. Para el segundo caso se debe encontrar un filtro centralizado para las situaciones donde las medidas se encuentran desfasadas. Se necesita una consideración especial para llevar a cabo una actualización, esta tesis tiene la ventaja de presentar la actualización secuencial.

2.6.2. Ecuaciones para múltiples sensores

El blanco de interés se describe por la ecuación dinámica lineal estándar.

$$x(k+1) = F(k)x(k) + v(k) \quad (2.2)$$

tomada por simplicidad sin el control de la entrada.

Para el caso sincronizado.

Se asume que son N_S sensores sincronizados. En un momento dado de muestreo es una medición por cada sensor, sin la incertidumbre de origen. Las medidas del sensor j , en el tiempo k son:

$$z(k, j) = H(k, j)x(k) + w(k, j) \quad j = 1, \dots, N_S \quad (2.3)$$

Los marcos de cada secuencia con ruido en las medidas tienen media cero, con procesos de ruido independiente en la ecuación 2.1, e independientes de sensor a sensor, con covarianzas.

$$E[w(k, j)w(l, i)'] = R(k, j)\delta_{ij}\delta_{kl} \quad (2.4)$$

Las dos técnicas se describen en la secuela para esta situación.

- Actualización Secuencial: La actualización se lleva a cabo con las medidas de un sensor para un tiempo, esto es visible en la ecuación 2.3

- Actualización Paralela: Las medidas de cada sensor para el mismo tiempo son acumuladas y se lleva a cabo una actualización simultánea.

Como se mencionó anteriormente el caso a emplear en esta tesis será la actualización secuencial.

2.6.3. Actualización Secuencial

La actualización secuencial entre los sensores para **medidas sincronizadas** es como se denota en la siguiente ecuación.

Nótese el **estado estimado** para el tiempo k y su covarianza como:

$$x(k|k, 0) \triangleq \hat{x}(k|k-1) \quad (2.5)$$

$$P(k|k, 0) \triangleq P(k|k-1) \quad (2.6)$$

Las actualizaciones con las medidas 2.2 en el tiempo k son:

$$\hat{x}(k|k, j) = \hat{x}(k|k-1) + W(k, j)v(k, j) \quad j = 1, \dots, N_S \quad (2.7)$$

donde:

$$v(k, j) \triangleq z(k, j) - H(k, j)\hat{x}(k|k, j-1) \quad (2.8)$$

$$S(k, j) = H(k, j)P(k|k, j-1)H(k, j)' + R(k, j) \quad (2.9)$$

$$W(k, j) = P(k|k, j-1)H(k, j)'S(k, j)^{-1} \quad (2.10)$$

$$P(k|k, j) = P(k|k, j-1) - W(k, j)S(k, j)W(k, j)' \quad (2.11)$$

Esto es posible por la ecuación 2.6 que establece que la medición a través del tiempo y los sensores son una secuencia en blanco. La actualización estimada final y la covarianza para el tiempo k son:

$$\hat{x}(k|k) = \hat{x}(k|k, N_S) \quad (2.12)$$

$$P(k|k) = P(k|k, N_S) \quad (2.13)$$

2.6.4. Filtro Kalman

Como se mencionó anteriormente los modelos de seguimiento son de especial interés para modelar un proceso donde se puedan reflejar todas las variables que intervienen es su dinámica. La forma más conveniente es la representación en variables de estado, aplicando este modelo ahora uno de los objetivos sería el cómo obtener información de la dinámica del proceso sin necesidad de medir todas las variables, sólo haciendo uso del conocimiento de su dinámica para poder inferir u observar algunas de ellas y por último, añadir a este modelo un tercer objetivo, que es intentar utilizarlo con mediciones con ruido dadas por algún tipo de incertidumbre. Utilizando los tres conceptos anteriores se aborda en esencia el Filtro Kalman. El tratamiento de este filtro esta ligado a dos aspectos de control: Uno es la obervación o estimación de estados internos de un sistema y por otro, teniendo esta misma observación, cuando las mediciones estan contaminadas por algún tipo de perturbación, [91].

Este filtro es el principal algoritmo para estimar sistemas dinámicos especificados en la forma de estado-espacio, (State-space) [92], la imagen representativa de este algoritmo y algunos términos se encuentran en el Apéndice C.

El filtro Kalman es un método de estimación cuyos parámetros se corrigen en cada iteración dependiendo del error de predicción que se tenga en la iteración anterior. Es un estimador lineal y óptimo desde el punto de vista de los mínimos cuadrados, ya que, en cada momento del tiempo, con base en la información disponible en el momento $t - 1$, actualiza la información adicional disponible en el momento t , de dichas estimaciones. En esta sección se explican los conceptos sobre los cuales se basa el filtro Kalman y se derivan sus ecuaciones. El filtro es muy poderoso en varios aspectos, ya que, apoya las estimaciones del pasado, presente e incluso los estados futuros y puede hacerlo incluso cuando la naturaleza precisa del sistema modelado es desconocida [93], [94], [95], [96].

El sistema físico se modela por un vector de estados x , llamado simplemente el estado y un conjunto de ecuaciones llamadas el modelo del sistema. El modelo del sistema es una ecuación de vectores que describe la evolución del estado en el tiempo. El tiempo de observación tiene la forma $t_k = t_0 + k\Delta T$, $k = 0, 1, \dots, N\Delta T$ es el intervalo de muestreo y x_k el estado $x(t_k)$.

El estado estimado desde t_k a t_{k+1} ignorando el proceso de ruido. Para un sistema con una matriz de transición $F(t_{k+1}, t_k)$, con un estado estimado $\hat{x}(k|k)$ y covarianza $P(k|k)$, la predicción es:

$$\hat{x}(k+1|k) = F(t_{k+1}, t_k)\hat{x}(k|k) \quad (2.14)$$

$$P(k+1|k) = F(t_{k+1}, t_k)P(k|k)F(t_{k+1}, t_k)' \quad (2.15)$$

El objetivo es actualizar el estado en el tiempo t_k usando las medidas que se encuentran en $z(k+1)$ para el tiempo t_{k+1} . Esto se realiza usando el estado estimado siguiente.

La inovación para el tiempo t_{k+1} y su covarianza son, con la notación estándar.

$$v(k+1) = z(k+1) - H(k+1)\hat{x}(k+1|k) \quad (2.16)$$

$$S(k+1) = H(k+1)P(k+1|k)H(k+1)' + R(k+1) \quad (2.17)$$

El incremento del filtro $W(k+1)$ puede ser calculado por la siguiente ecuación

$$W(k+1) = P(k+1|k)H(k+1) + S(k+1)^{-1} \quad (2.18)$$

y el nuevo estado de actualización para el tiempo t_k , (el cual ahora se ha incorporado a las medidas para el tiempo t_{k+1}) y su covarianza son:

$$\hat{x}(k|k+1) = \hat{x}(k|k) + W(k+1)v(k+1) \quad (2.19)$$

$$P(k|k+1) = P(k|k) - W(k+1)S(k+1)W(k+1)' \quad (2.20)$$

Para cualquier estado k donde todo estado anterior $k-1$ es resumido por el estado estimado $\hat{x}(k|k)$ y la covarianza asociada $P(k|k)$.

La columna izquierda del siguiente diagrama de flujo muestra la evolución verdadera del sistema desde el tiempo k para el tiempo $k+1$ con una entrada $u(k)$ y un ruido del proceso $v(k)$, las medidas siguientes vienen del nuevo estado y el ruido $w(k+1)$.

El ciclo del estado estimado, (columna intermedia del diagrama), consiste de:

- Estado y medidas de predicción, (también llamadas “tiempo de actualización”).
- Estado actualizado, (tambien llamado “medidas actualizadas”).

La entrada conocida $x(k)$, (control o movimiento de la plataforma o señalización del sensor), es usado por el estimador del estado para obtener la predicción del estado del siguiente tiempo. El estado actualizado requiere del filtro de ganancia, obtenido en el curso de los cálculos de la covarianza en la columna derecha.

2.7. Asociación

En esta sección se darán a conocer los procesos inmersos en la asociación de datos. Una vez analizados cada uno de ellos se procederá a dar a conocer los algoritmos empleados en esta tesis.

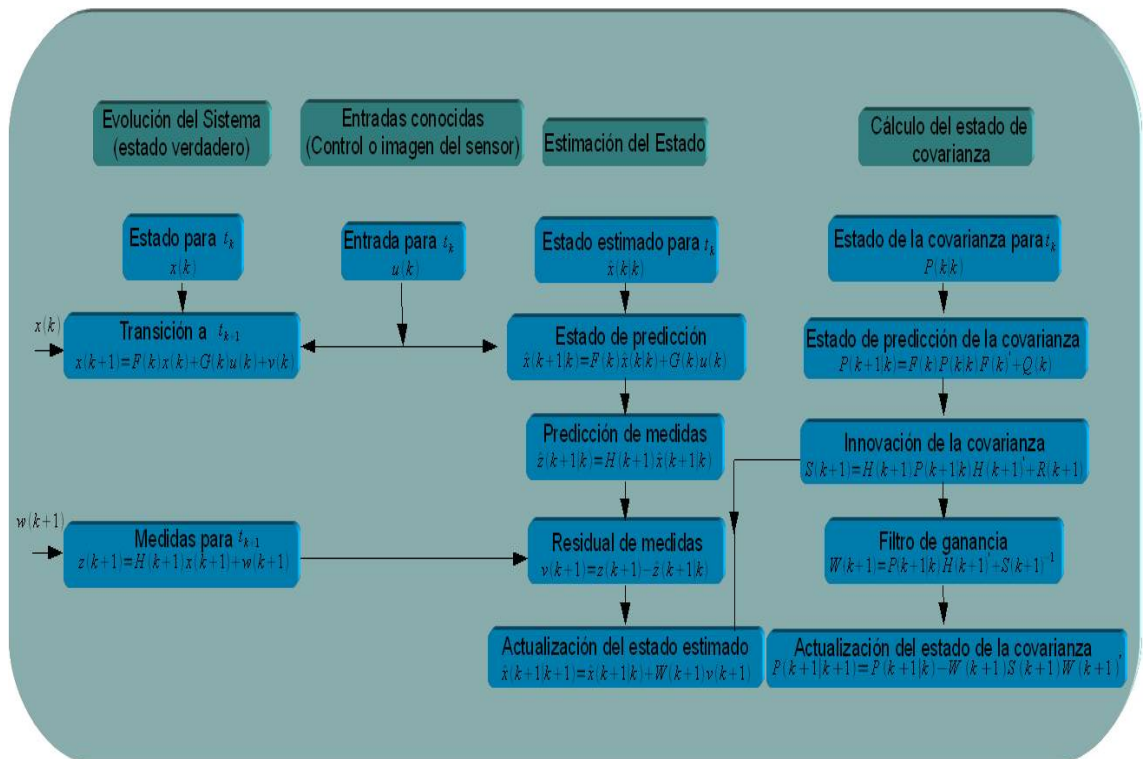


Figura 2.6: Un ciclo en el estado de Estimación del Filtro Kalman

2.7.1. Región de Validación

En los sistemas de seguimiento un paso a analizar es la detección de los objetivos mediante la selección de las medidas, este proceso asocia las medidas incorporando una **región de validación o asociación**, (gate).

La cual se define como la región en el espacio de medidas alrededor de las medidas estimadas de las cuales la medida verdadera se puede tener dentro de la validación, en vista de todas las incertidumbres existientes con la probabilidad alta de tener una certera estimación.

Una medida que se encuentre en el gate ó umbral será tomada como una candidata a asociarse a un rastreo generado existente. Algunas formas de generar esta región de asociación se ven en los algoritmos PDA y JPDA, empleando el filtro Kalman.

El vector de medidas debe de incluir como componentes todas las variables que mantienen estas características:

- Contiene información sobre el estado del objetivo.
- Discrimina las medidas indeseables.

Considerando que el objetivo existe en un rastreo o track, es decir, su filtro ha sido analizado, entonces de acuerdo a la fórmula de predicción de medidas $\hat{z}(k+1|k)$ y la covarianza asociada $S(k+1)$. Se asume lo siguiente:

La medida verdadera condicionada en el tiempo $t-1$ es normalmente distribuida con su **función de densidad de probabilidad** dada por:

$$p[z(k+1)|Z^k] = \mathcal{N}[z(k+1); \hat{z}(k+1|k), S(k+1)] \quad (2.21)$$

Donde la medida verdadera puede caer en la siguiente región

$$\mathcal{V}(k+1, \gamma) = z : [z - \hat{z}(k+1|k)]' S(k+1)^{-1} [z - \hat{z}(k+1|k)] \leq \gamma \quad (2.22)$$

con la probabilidad determinada por la **región de umbralización** γ .

La región definida por 2.21 es llamada región de validación, denotada por el símbolo \mathcal{V} ó región de asociación. También conocido como elipse ó elipsoide de concentración de probabilidad.

El umbral γ se obtiene de las tablas de la distribución Chi - cuadrado con un número de grados de libertad igual a la dimensión n_z de las medidas.

2.7.2. Problemas de la Asociación: Un Objetivo en Interferencia

El procedimiento de validación delimita la región del espacio de medidas donde la información procesada podría encontrar la medida de interés del objetivo. A pesar de tener más de una detección, es decir, varias medidas podrían ser encontradas en la región de validación.

Las medidas que se encuentran fuera de la región de validación pueden ser ignoradas: Si estas se encuentran “bastante lejos”, por lo tanto es muy poco probable que se tenga un origen en el objetivo de interés. Si la probabilidad del umbral esta cerca de la unidad⁴ y el modelo usado para obtener el umbral es correcto.

El problema del seguimiento de un solo objetivo en un ambiente desordenado ó con falsas medidas, considera la situación donde hay varias posibles medidas en la región de validación de un objetivo. El conjunto de **medidas válidas** consiste de:

- Medidas correctas (si estas son detectadas y estan dentro del umbral).
- Medidas indeseables (falsas alarmas originadas).

El ejemplo donde se presenta este conjunto de medidas se muestra en la siguiente figura. La región de validación es una elipse centrada para las medidas estimadas $\hat{z}(k+1|k)$. Los parámetros de la elipse son determinados por la matriz de innovación de la covarianza $S(k+1)$.

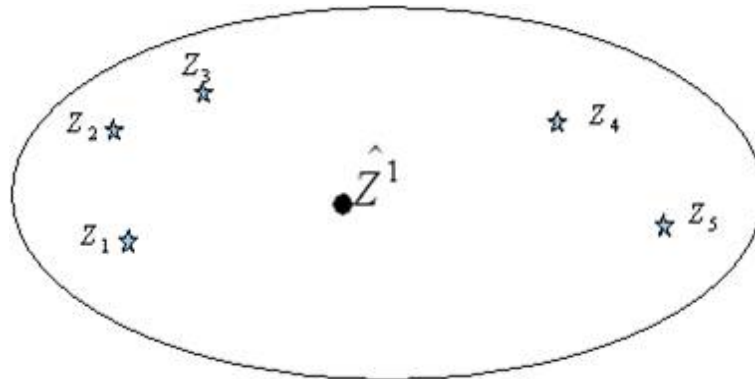


Figura 2.7: Varias medidas, dentro de una región de validación y un solo objetivo válido \hat{Z}^1

La implicación de la hipótesis de que existe un objetivo único es que las medidas no deseadas constituyen una interferencia aleatoria. Se dice entonces que las medidas que se encuentran dentro de la región de validación tienen una probabilidad de tener su origen en el objetivo de interés, sin embargo, se asume que solo una ha sido generada por el objetivo.

2.7.3. Problemas de la Asociación: Múltiples Objetivos en Interferencia

En el caso donde hay varios objetivos en la misma vecindad también con interferencia es mucho más complicado.

⁴Empleando la función de densidad de probabilidad

La siguiente figura ilustra tal caso dado en un tiempo, con medidas estimadas para dos objetivos considerados denotados como: \hat{Z}^1 y \hat{Z}^2 .

En esta figura las siguientes medidas originadas son posibles:

- Z_1 desde el objetivo \hat{Z}^1 o en interferencia.
- Z_2 desde ambos objetivos \hat{Z}^1 o \hat{Z}^2 o en interferencia.
- Z_4 y Z_5 desde el objetivo \hat{Z}^2 o en interferencia.

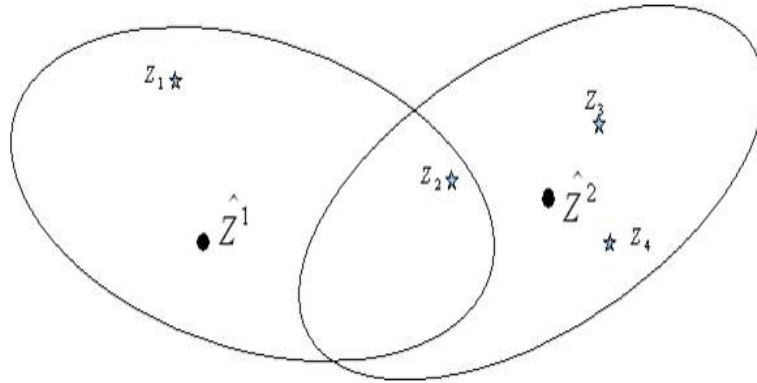


Figura 2.8: Dos objetivos \hat{Z}^1 y \hat{Z}^2 , con una medida verdadera cada uno y la intersección de sus regiones de validación

Sin embargo, si Z_2 fue originado desde \hat{Z}^2 entonces es muy probable que Z_1 haya sido originado por el objetivo \hat{Z}^1 . Esto ilustra la interdependencia de la asociación en una situación donde una **Interferencia persistente** (objetivo vecino) se presenta en adición a la interferencia aleatoria, (Interferencia ó clutter).

Hasta este punto se asume que una medida podría tener su origen desde un objetivo o desde un punto de la interferencia. Sin embargo, cualquier sistema de procesamiento tiene una inherente capacidad de **resolución finita**, una posibilidad adicional tiene que ser considerada.

Z_2 podría ser el resultado de la fusión de la detección de dos objetivos, a esto se le llama una **medida sin resolver**.

2.7.4. Modelo del Objetivo

Si se asume que el objetivo esta desplazándose en un espacio bidimensional y sus imágenes son modeladas como independientes entre sus coordenadas. El estado de la covarianza por cada coordenada actualizada de acuerdo a la ecuación modificada de Ricatti [97] con un “indicador de detección” es como sigue:

$$P(k+1|k) = FP(k|k)F' + Q \quad (2.23)$$

$$P(k+1|k+1) = P(k+1|k) - \delta(k+1)P(k+1|k)H'S(k+1)^{-1}HP(k+1|k) \quad (2.24)$$

Sobre F el bloque diagonal de la matriz de transición consiste en dos bloques, Q es el bloque diagonal de proceso del ruido de la covarianza.

$$S(k+1) = HP(k+1|k)H' + R \quad (2.25)$$

Es la matriz de innovación de la covarianza para el tiempo $k+1$, H es la matriz de medidas, R es la matriz de covarianza del ruido de las medidas con los elementos diagonales r y

$$\delta(k) = \begin{cases} 1 & \text{Si la detección esta en el umbral para } k, \\ 0 & \text{En otro caso} \end{cases} \quad (2.26)$$

es el **indicador de la detección** para el tiempo k .

2.7.5. Modelo para el Seguimiento de un solo Objetivo en Interferencia

Las ecuaciones de estado y medidas se asumen lineales que con ruido blanco añadido a la normalización media cero con las covarianzas conocidas.

El estado del objetivo de interes de dimensión n_x , se asume involucrado en el tiempo de acuerdo a la ecuación.

$$x(k+1) = F(x)x(k) + v(k) \quad (2.27)$$

con las medidas verdaderas de dimensión n_x , dada por

$$z(k) = H(k)x(k) + w(k) \quad (2.28)$$

donde $v(k)$ y $w(k)$ son normalizaciones media cero mutuamente independientes con secuencias de ruidos Gaussiano de las covarianzas conocidas $Q(k)$ y $R(k)$, respectivamente.

En este algoritmo se asume que el seguimiento esta inicializado en un estado inicial estimado y una covarianza válida.

La región de validación es actualizada para cualquier tiempo de la secuencia alrededor de las medidas y posibles medidas que caen en esta región. Estas difieren de como estas medidas son usadas o no en la estimación del estado del objetivo. Todas las variables que tienen información útil, para diferenciar la medida correcta de una incorrecta, se asume que son incluidas en el vector de medidas.

2.7.6. Algoritmos de Seguimiento para un solo Objetivo en Interferencia

Se presenta cuando el estado de estimación de un solo objetivo ha usado medidas de origen inciertas. El estado del objetivo es observado con la probabilidad de detección posiblemente menor que la unidad y en presencia del desorden, el cual incrementa el detectar falsas alarmas.

Las trayectorias o rastreos se asumen inicializados, es decir, el **mantenimiento del rastreo**. Para el tiempo de validación o la región de asociación se pueden encontrar una o más medidas.

Asumiendo que la información de intensidad de la señal esta disponible, una alternativa es seleccionar la medida más fuerte entre las demás medidas válidas. Algo parecido pasa con el algoritmo **NNKF**.

A cualquier medida que pueda ser válida, originada por el objetivo, se sugiere que todas las medidas desde la región de validación deben ser usadas con alguna moda.

El enfoque Bayesiano llamado **Probabilistic Data Association (PDA)**, asocia la probabilidad de todos los “vecinos” de cada medida validada en el tiempo actual, para el objetivo de interés. Esta información de probabilidad es usada en un filtro de seguimiento llamado **filtro PDA** que cuenta con las medidas originadas por la incertidumbre.

2.7.6.1. Algoritmo No Bayesiano: Nearest Neighbor Kalman Filter

El algoritmo consiste en los siguientes pasos:

- Validación de medidas usando la ecuación 2.21.
- Selección de una de las medidas válidas.
- Actualización de un estado del objetivo como si una de las medidas fuera correcta, es decir, en la manera del Filtro de Kalman.

Una vez realizado lo anterior se selecciona la medida cercana mediante la estimación de la distancia de la medida de esta.

$$D(z) = [z - \hat{z}(k+1|k)]' S(k+1)^{-1} [z - \hat{z}(k+1|k)] = v' S(k+1)^{-1} v \quad (2.29)$$

donde S es la matriz de innovación de la covarianza, como se muestra en el diagrama.

2.7.6.2. Algoritmo Bayesiano: Probabilistic Data Association

Este algoritmo calcula la asociación de probabilidades de cada medida validada del objetivo de interés en el tiempo actual. Esta información de probabilidad (Bayesiana), es usada por el filtro de seguimiento llamado **filtro PDA** que cuenta las medidas originadas por la incertidumbre, es decir:

Se asume que:

- Solo existe un único objetivo de interés 2.26 y 2.27.
- El rastreo o trayectoria ha sido inicializado.
- La información anterior sobre el objetivo es resumida aproximadamente por:

$$p[x(k)|Z^{k-1}] = \mathcal{N}[x(k); \hat{x}(k|k-1), P(k|k-1)] \quad (2.30)$$

- Para cada tiempo se crea una región de validación como en 2.21
- Entre varias posibles medidas válidas, para más de una de ellas, se puede tener un objetivo originado de acuerdo a la ecuación 2.27, si el objetivo fué detectado y la medida correcta cae dentro de la región de validación.
- Las medidas restantes se asumen son falsas alarmas o interferencia y son modeladas de manera **independiente e idénticamente distribuidas** con una distribución espacial uniforme.
- La detección del objetivo ocurre independientemente sobre el tiempo con una probabilidad de detección conocida P_D .

2.7.6.2.1. Enfoque PDA

El PDA usa la descomposición de la estimación del origen de cada elemento de la última serie de medidas validadas, denotadas como:

$$Z(k) = \{z_i(k)\}_{i=1}^{m(k)} \quad (2.31)$$

donde $z_i(k)$ es la i -ésima medida validada y $m(k)$ es el número de medidas en la región de validación para el tiempo k .

El conjunto acumulativo, (secuencia) de medidas es:

$$Z^k = \{Z(j)\}_{j=1}^k \quad (2.32)$$

El algoritmo óptimo agrega la descomposición con respecto a cada secuencia de medidas desde el conjunto 2.31. En orden de agregar una descomposición de las últimas medidas solamente, se dan las realizadas por 2.29.

2.7.6.2.2. Medidas de Validación

Siguiendo la ecuación 2.29, la región de validación es de forma elíptica

$$\mathcal{V}(k, \gamma) = \{z : [z - \hat{z}(k|k-1)]' S(k)^{-1} [z - \hat{z}(k|k-1)] \leq \gamma\} \quad (2.33)$$

donde γ es el umbral de validación y

$$S(k) = H(k)P(k|k-1)H(k)' + R(k) \quad (2.34)$$

es la innovación de la covarianza correspondiente para las medidas verdaderas.

2.7.6.2.3. El Estado de Estimación

En vista de asumir los eventos de asociación:

$$\theta_i(k) \begin{cases} \{z_i(k) \text{ es el objetivo originado por las medidas}\} & i = 1, \dots, m(k) \\ \{\text{ninguna de las medidas es un objetivo originado}\} & i = 0 \end{cases} \quad (2.35)$$

son mutuamente exclusivas y exhaustivas para $m(k) \geq 1$.

Usando el teorema de probabilidad total sobre los eventos de la media condicional del estado para el tiempo k puede ser escrito como:

$$\begin{aligned} \hat{x}(k|k) &= E[x(k)|Z^k] \\ &= \sum_{i=0}^{m(k)} E[x(k)|\theta_i(k), Z^k] P\{\theta_i(k)|Z^k\} \\ &= \sum_{i=0}^{m(k)} \hat{x}_i(k|k) \beta_i(k) \end{aligned} \quad (2.36)$$

donde $\hat{x}_i(k|k)$ es el estado actualizado condicionado sobre el evento de la i -ésima medida válida correcta y

$$\beta_i(k) P \triangleq \{\theta_i(k)|Z^k\} \quad (2.37)$$

es la probabilidad condicional de este evento, la **probabilidad de asociación**, obtenida desde el procedimiento PDA.

El estado estimado condicionado sobre las medidas i es correcto:

$$\hat{x}_i(k|k) = \hat{x}(k|k-1) + W(k)v_i(k) \quad i = 1, \dots, m(k) \quad (2.38)$$

donde la innovación correspondiente es

$$v_i(k) = z_i(k) - \hat{z}_i(k|k-1) \quad (2.39)$$

La ganancia $W(k)$ es la misma como en el filtro estandar

$$W(k) = P(k|k-1)H(k)'S(k)^{-1} \quad (2.40)$$

tal que, la condicional sobre $\theta_i(k)$, no es una medida de origen incierto.

Para $i = 0$, es decir, si ninguna de las medidas es correcta ó si no es una medida válida, ($m(k) = 0$), se tiene

$$\hat{x}_0(k|k) = \hat{x}(k|k-1) \quad (2.41)$$

2.7.6.2.4. Estado de la Covarianza Actualizado

Combinando 2.37, 2.40 y 2.35, se obtiene el estado actualizado de la ecuación del PDA

$$\hat{x}(k|k) = \hat{x}(k|k-1) + W(k)v(k) \quad (2.42)$$

donde la innovación combinada es

$$v(k) = \sum_{i=1}^{m(k)} \beta_i(k)v_i(k) \quad (2.43)$$

La covarianza asociada con el estado actualizado es

$$P(k|k) = \beta_0(k)P(k|k-1) + [1 - \beta_0(k)]P^c(k|k) + \tilde{P}(k) \quad (2.44)$$

donde la covarianza del estado actualizado con la medida correcta es

$$P^c(k|k) = P(k|k-1) - W(k)S(k)W(k)' \quad (2.45)$$

y el término de la **extensión de la innovación** esta dado por:

$$\tilde{P}(k) \triangleq W(k) \left[\sum_{i=1}^{m(k)} \beta_i(k)v_i(k)v_i(k)' - v(k)v(k)' \right] W(k)' \quad (2.46)$$

2.7.6.2.5. Ecuaciones de Predicción

La predicción del estado y las medidas para $k + 1$ esta dada en el filtro estándar de Kalman

$$\hat{x}(k+1|k) = F(k)\hat{x}(k|k) \quad (2.47)$$

$$\hat{z}(k+1|k) = H(k+1)\hat{x}(k+1|k) \quad (2.48)$$

La covarianza del estado estimado es similar

$$P(k+1|k) = F(k)P(k|k)F(k)' + Q(k) \quad (2.49)$$

donde $P(k|k)$ esta dado por 2.43.

La innovación de la covarianza, para las medidas correctas se obtiene con el filtro estándar

$$S(k+1) = H(k+1)P(k+1|k)H(k+1)' + R(k+1) \quad (2.50)$$

2.7.6.2.6. Probabilistic Data Association

Para evaluar las probabilidades de asociación, la condicionante se descompone dentro de los datos anteriores Z^{k-1} y el ultimo dato $Z(k)$.

Ya que una inferencia de probabilidad puede ser hecha en ambos tanto en el numero de medidas que se encuentren dentro de la región de validación como bien sobre su localización, explicitamente la formula encontrada es

$$\beta_i(k) = P\{\theta_i(k)|Z^k\} = P\{\theta_i(k)|Z(k), m(k), Z^{k-1}\} \quad (2.51)$$

usando la fórmula de Bayes se reescribe como

$$\beta_i(k) = \frac{1}{c} p[Z(k)|\theta_i(k), m(k), Z^{k-1}] P\{\theta_i(k)|m(k), Z^{k-1}\} \quad i = 0, 1, \dots, m(k) \quad (2.52)$$

La PDF ⁵ de las medidas correctas, con el factor que cuenta la probabilidad del gating, P_G delimita la densidad normal del umbral de validación es

$$p[z_i(k)|\theta_i(k), m(k), Z^{k-1}] = P_G^{-1} \mathcal{N}[z_i(k); \hat{z}(k|k-1), S(k)] = P_G^{-1} \mathcal{N}[v_i(k); 0, S(k)] \quad (2.53)$$

La PDF desde 2.51 es entonces

$$p[Z(k)|\theta_i(k), m(k), Z^{k-1}] \begin{cases} V(k)^{-m(k)+1} P_G^{-1} \mathcal{N}[v_i(k); 0, S(k)] & i = 1, \dots, m(k) \\ V(k)^{-m(k)} & i = 0 \end{cases} \quad (2.54)$$

Las probabilidades de los eventos de asociación condicionados solo sobre el número de medidas validadas son:

$$= \begin{cases} \frac{1}{m(k)} P_D P_G [P_D P_G + (1 - P_D P_G) \frac{\mu_F[m(k)]}{\mu_F[m(k)-1]}]^{-1} & i = 1, \dots, m(k) \\ (1 - P_D P_G) \frac{\mu_F[m(k)]}{\mu_F[m(k)-1]} [P_D P_G + (1 - P_D P_G) \frac{\mu_F[m(k)]}{\mu_F[m(k)-1]}]^{-1} & i = 0 \end{cases} \quad (2.55)$$

⁵En teoría de la probabilidad, la función de densidad de probabilidad, función de densidad, o, simplemente, la densidad de una variable aleatoria continua es una función, usualmente denominada $f(x)$ que describe la densidad de la probabilidad en cada punto del espacio de tal manera que la probabilidad de que la variable aleatoria tome un valor dentro de un determinado conjunto sea la integral de la función de densidad sobre dicho conjunto. Es decir, la probabilidad de que una variable aleatoria X tome un valor determinado entre dos números reales a y b coincide con el área encerrada por la función [98]

$$F(x) = \frac{e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}}{\sigma\sqrt{2\pi}}$$

, **función de densidad de la probabilidad**, entre los puntos a y b , es decir:

$$P(a \leq X \leq b) = \int_a^b f(x) dx$$

donde $\mu_F(m)$ es la función de masa de probabilidad⁶, del número de medidas falsas, (interferencia) en la región de validación.

El modelo usado para la FMP $\mu_F(m)$ del volumen de interés V , para PDA es el modelo parametrizado de **Poisson** con una densidad espacial de certeza λ

$$\mu_F(m) = e^{-\lambda V} \frac{(\lambda V)^m}{m!} \quad (2.56)$$

usando Poisson en 2.54 se tiene

$$\gamma_i[m(k)] = \begin{cases} P_D P_G [P_D P_G m(k) + (1 - P_D P_G) \lambda V(k)]^{-1} & i = 1, \dots, m(k) \\ (1 - P_D P_G) \lambda V(k) [P_D P_G m(k) + (1 - P_D P_G) \lambda V(k)]^{-1} & i = 0 \end{cases} \quad (2.57)$$

2.7.6.2.7. PDA Paramétrico

Usando 2.53 y 2.56 con la expresión explícita de la FDP Gaussiana dentro de 2.51, después de algunas cancelaciones, la ecuación final del PDA paramétrico con el **modelo de interferencia de Poisson**

$$\beta_i(k) = \begin{cases} \frac{e_i}{b + \sum_{j=1}^{m(k)} e_j} & i = 1, \dots, m(k) \\ \frac{b}{b + \sum_{j=1}^{m(k)} e_j} & i = 0 \end{cases} \quad (2.58)$$

donde

$$e_i \triangleq e^{-\frac{1}{2} v_i(k)' S(k)^{-1} v_i(k)} \quad (2.59)$$

$$b \triangleq \lambda |2\pi S(k)|^{\frac{1}{2}} \frac{1 - P_D P_G}{P_D} \quad (2.60)$$

2.7.7. Algoritmo Bayesiano: Joint Probabilistic Data Association

La estimación Joint Probabilistic Data Association es una extensión del método anterior.

En este algoritmo se asumen los siguientes puntos:

- Se tiene un número de objetivos establecidos en interferencia.
- Las medidas de un objetivo pueden caer en la región de validación de un objetivo vecino. Esto puede suceder si hay varios periodos de muestreo y hechos como una interferencia constante.

⁶Se le llama **Función de masa de probabilidad** o FMP de una variable aleatoria discreta X a la probabilidad de que X tome un valor $x \in \mathfrak{R}$, $P(x) = P(X = x) = P(\{w \in \Omega : X(w) = x\})$ [99]

- El estado anterior puede ser resumido por una eficiente estadística estimada, (media condicional aproximada) estados estimados y covarianzas para cada objetivo.
- Los estados se asumen Gaussianos distribuidos por sobre las medias y las covarianzas.
- Cada objetivo tiene una dinámica y un modelo de medidas como en 2.26 y 2.27. Los modelos para cada objetivo no tienen por que ser los mismos.

El modelado del JPDA asume que todas las medidas son incorrectas como una interferencia aleatoria con una distribución espacial uniforme. El desempeño del filtro PDA se degrada significativamente cuando un objetivo vecino da lugar a una interferencia constante.

2.7.7.0.8. Enfoque JPDA

Para poder desarrollar el algoritmo JPDA, se deben tomar en cuenta las siguientes consideraciones:

- Las medidas nuevas y las probabilidades de asociación de los objetivos son calculadas conjuntamente.
- Las probabilidades de asociación son calculadas solo por el último conjunto de medidas.
- Una matriz de validación esta definida con elementos que indican si las medidas j caen en el umbral de validación de un objetivo t .
- La asociación creible de eventos se obtiene asumiendo:
 - Una fuente, (objetivo o interferencia), para una medida, y
 - Más de una medida por objetivo
- Como sucede con el PDA, hay una función de masa de probabilidad, ya sea paramétrica o no paramétrica, en este caso se empleará la ecuación paramétrica.

2.7.7.0.9. Estado de estimación

Se tienen dos opciones de estados de estimación:

- Los estados del objetivo condicionado sobre las observaciones anteriores se asumen independientes. Entonces bajo esta evaluación el **filtro Joint Probabilistic Data Association, (JPDAF)** se desvincula de la estimación de los objetivos.

En este caso uno necesita las probabilidades de asociación marginal, el cual se obtienen desde las probabilidades adjuntas, resumiendo sobre todos los eventos adjuntos en los cuales ocurre el evento marginal de interés.

- Los estados de los objetivos dados en el pasado son correlacionados, estos son caracterizados por medias, covarianzas así como las covarianzas cruzadas.

Esto lleva a cabo la estimación para los objetivos bajo evaluación, el **JPDA**.

2.7.7.0.10. Características especiales del JPDA

La evaluación de la probabilidad condicional de los eventos de asociación conjuntos pertenecientes al tiempo actual k , (el tiempo indexado con k es omitido por simplicidad donde ello no cause confusión), se añade como:

$$\theta = \bigcap_{j=1}^m \theta_{jt_j} \quad (2.61)$$

donde:

- θ_{jt} es el evento de medidas j originadas desde el objetivo t , $j = 1, \dots, m$; $t = 0, 1, \dots, N_T$
- t_j es el índice del objetivo para el cual las medidas j son asociadas en el evento bajo su consideración.
- N_T es el número conocido de objetivos

2.7.7.0.11. La Matriz de Validación

Definiendo la matriz de validación se tiene

$$\Omega = [w_{jt}] \quad j = 1, \dots, m; t = 0, 1, \dots, N_T \quad (2.62)$$

con elementos binarios que indican si las medidas j caen en el umbral de validación del objetivo t . El índice $t = 0$ significa que “ no se tiene ninguno de los objetivos ” y corresponde a la columna de Ω , que tiene todas las unidades desde que cada medida podría ser originada desde la interferencia o una falsa alarma.

2.7.7.0.12. JPDA Paramétrico

Como en el caso del PDA, JPDA tiene dos versiones de acuerdo al modelo usado para la PMF $\mu_F(\phi)$ del numero de falsas medidas.

El JPDA paramétrico usa la PMF de Poisson

$$\mu_F(\phi) = e^{-\lambda V} \frac{(\lambda V)^\phi}{\phi!} \quad (2.63)$$

el cual requiere de espacio de densidad λ de falsas medidas.

La probabilidad de asociación conjunta del JPDA paramétrico es

$$P\{\theta(k)|\mathcal{Z}^k\} = \frac{\lambda^\phi}{c_1} \prod_j \{f_{t_j}[z_j(k)]\}^{T_j} \prod_j (P_D^t)^{\delta_t} (1 - P_D)^{(1-\delta_t)} \quad (2.64)$$

donde c_1 es una constante de normalización apropiada.

Desde que $m(k)$ es un número fijo, se define una constante de normalización

$$c_2 \triangleq c_1 \lambda^{-m(k)} \quad (2.65)$$

La ecuación 2.63 puede ser reescrita como:

$$P\{\theta(k)|\mathcal{Z}^k\} = \frac{1}{c_2} \prod_j \{\lambda^{-1} f_{t_j}[z_j(k)]\}^{T_j} \prod_j (P_D^t)^{\delta_t} (1 - P_D)^{(1-\delta_t)} \quad (2.66)$$

Capítulo 3

Implementación de los Algoritmos de Fusión basados en la Asociación de Datos

La ciencia sola no es capaz de responder todas las preguntas y, pese a su desarrollo, jamás lo será. Claude Lévi Strauss

3.1. Introducción

En este capítulo se presenta la propuesta del modelo de fusión de datos el cual esta diseñado para fusionar datos provenientes de sensores visuales además del proceso de detección, la generación de la matriz homográfica y la utilización del filtro Kalman en los algoritmos de asociación NNFK, PDA y JPDA.

3.2. Modelo de Fusión

El modelo inicia con la entrada de datos que porporcionan los sensores visuales, es decir, el algoritmo de detección ha encontrado puntos en el centro de proyección de las cámaras por lo tanto se realizan las operaciones pertinentes para el trasladado de los puntos a un solo plano y posteriormente se procede a realizar la fusión de los datos con alguno de los algoritmos de asociación.

Es importante mencionar que este modelo se encuentra a nivel características del modelo de fusión de imágenes. En las secciones siguientes se explica a detalle cada proceso.

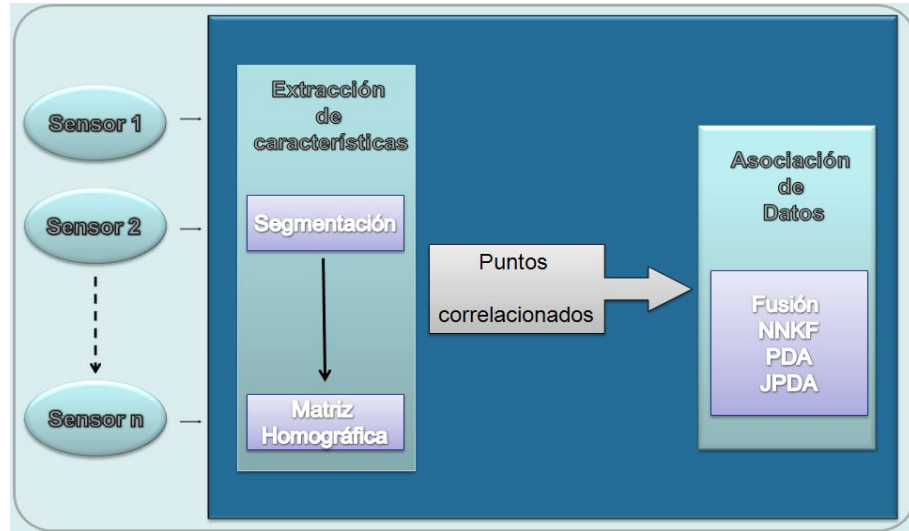


Figura 3.1: Modelo de Fusión de Imágenes

3.3. Detección

La detección es un proceso fundamental para el seguimiento de objetos o personas, es aquí donde se obtiene la mayor cantidad de información posible de la imagen. La segmentación es el algoritmo encargado de realizar la detección, ya que en esta etapa del procesamiento de la imagen se logran detectar las regiones de interés además de hacer un estimado de la localización de las posiciones $x's$, $y's$ de cada región.

3.3.1. Proceso de la Segmentación



Figura 3.2: Escenario base



Figura 3.3: Objetivos nuevos en la escena

El proceso de la segmentación se empieza con la descomposición de la imagen del canal RGB a tres canales ó imágenes, después se procede a transformar los canales al

espacio de colores YIQ ¹, el proceso en sí convierte los tres canales RGB a uno solo con el objetivo de eliminar en gran parte los problemas de brillo, contraste o ruido encontrados en las escenas, este proceso utiliza la siguiente ecuación.

$$Y = 0,299 * Red + 0,587 * Green + 0,114 * Blue. \quad (3.2)$$

Una vez realizada dicha transformación se procede a realizar la sustracción del marco base con los marcos temporales de cada secuencia para poder detectar los objetos.



Figura 3.4: Sustracción de imágenes

Denotando $G1$ como la imagen de fondo y $G2_i$, como la secuencia de imágenes de cada escenario.

$$G' = (G1 - G2_i) * Mult + Add \quad (3.3)$$

Entre las operaciones utilizadas para la segmentación esta el filtro de Sobel, el cual calcula la primera derivada de una imagen para posteriormente realizar una umbralización y realizar la detección de los bordes. El filtro tiene las máscaras siguientes:

$$A = \begin{bmatrix} 1,0 & 2,0 & 1,0 \\ 0,0 & 0,0 & 0,0 \\ -1,0 & -2,0 & -1,0 \end{bmatrix} \quad (3.4)$$

¹ El modelo YIQ fué diseñado para obtener la ventaja del sistema visual humano, que es más sensible a cambios en la reflectancia que a cambios en el matiz o saturación.

Donde Y es la reflexión y la información del color esta dada por I y Q. I significa en fase (en inglés: in-fase), mientras que Q significa cuadratura (en inglés: quadrature). La principal ventaja del modelo YIQ es la reflectancia (Y) y la información de color (I y Q), se pueden tratar por separado. Por lo tanto, la importancia de esta separación es la componente de reflectancia de una imagen que puede ser procesada sin afectar a su contenido de color [100] [101]

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ 0,596 & -0,275 & -0,321 \\ 0,212 & -0,523 & -0,311 \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.1)$$

$$B = \begin{bmatrix} 1,0 & 0,0 & -1,0 \\ 2,0 & 0,0 & -2,0 \\ 1,0 & 0,0 & -1,0 \end{bmatrix} \quad (3.5)$$

Una vez detectados los bordes se procede a trabajar con las regiones, con el objetivo de detectar regiones que no se pudieron encontrar por el filtro de Sobel.

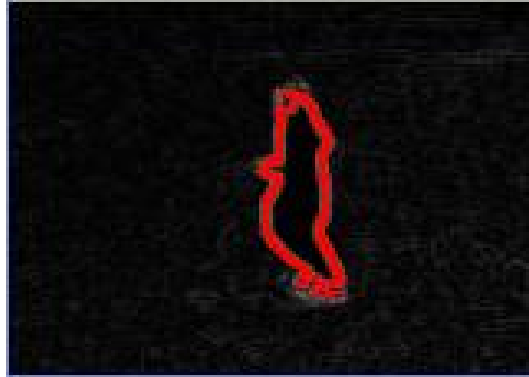


Figura 3.5: Filtro Sobel para la detección de Bordes



Figura 3.6: Umbralización para la detección de las Regiones

Después se procede a obtener las posiciones $x's, y's$ de cada región y verificar si estas se mantienen dentro del centro de proyección de los escenarios para realizar la transformación homográfica.

3.4. Plano Homográfico

Los escenarios utilizados en esta tesis fueron obtenidos del proyecto CAVIAR² [102], los cuales se grabaron en un pasillo de un centro comercial, que en este caso, es el centro de proyección, las secuencias fueron grabadas por dos cámaras posicionadas en distinto ángulo y en diferente lugar, pero proyectadas al mismo centro de proyección, por lo tanto, por cada secuencia de imágenes se tienen dos perspectivas distintas.

El proyecto CAVIAR proporciona los marcos base que se pueden correlacionar de un escenario a otro, para esto un escenario se ha denominado **Corredor** y el otro **Frontal**.

Puntos correlacionados del escenario Corredor:

Debe de tomarse en cuenta si se tienen menos de 4 pares de puntos (Px, Py, PW) , (Qx, Qy, QW) , no existe una solución única, si se presentan exactamente 4 pares de la matriz H se transforma en exactamente la matriz deseada y si hay más de 4 pares de puntos dados la operación intentará minimizar el error de transformación.

²CAVIAR es un proyecto financiado por la Comisión Europea para desarrollar un sistema de visión multisensor) de la comunicación de eventos pico.

Punto	(Col,Fil) (pixels)	(X,Y) (cm)
1	(91,163)	(000,975)
2	(241,163)	(290,975)
3	(98,266)	(000,-110)
4	(322,265)	(290,-110)

Tabla 3.1: Puntos del escenario Corredor.



Punto	(Col,Fil) (pixels)	(X,Y) (cm)
5	(60,153)	(000,000)
6	(359,153)	(000,975)
7	(50,201)	(382,098)
8	(167,200)	(382,878)

Tabla 3.2: Puntos del escenario Frontal.



Donde los cuatro puntos del escenario Corredor son dados por la matriz P y los puntos del escenario Frontal son aquellos que se encuentran en la matriz Q lo que se obtenga de ellos será la transformación homográfica y generación de la matriz H .

$$H \times \begin{bmatrix} P_x \\ P_y \\ P_w \end{bmatrix} = \begin{bmatrix} Q_x \\ Q_y \\ Q_w \end{bmatrix} \quad (3.6)$$

La imagen siguiente muestra la relación entre las filas y columnas y el eje de coordenadas ($x's, y's$).

Si los puntos a transformar se especifican en las coordenadas estándar las coordenadas fila deben pasar a ser los puntos P_x y las coordenadas columna en los puntos P_y . Esto es necesario para obtener un sistema de coordenadas de la imagen. En particular esto asegura que las rotaciones se realizan en la dirección correcta.

La matriz generada en 3.6 tiene nueve elementos pero solo ocho son independientes. Aunque en muchas ocasiones se fija la escala tomando el valor de $H_x = 1$, esto no es siempre correcto ya que en algunos casos el verdadero valor de H_x puede ser 0. Como se mencionó una proyección entre planos puede ser calculada a partir de la correspondencia entre cuatro puntos. Ahora al contrario de las afinidades no es posible distinguir entre proyecciones que preservan o invierten la orientación.

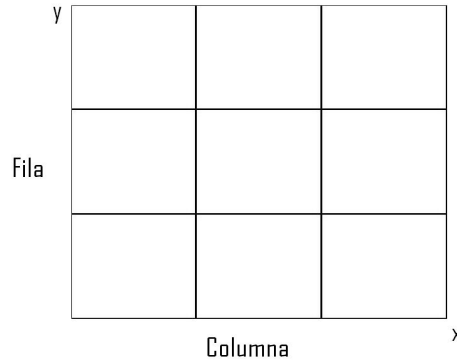


Figura 3.7: Imagen dividida en Filas y Columnas, tomando en cuenta que Columna es el eje de las x' s y Fila es el eje de las y' s

Para este caso en particular la transformación se hará del escenario Frontal al Corredor, es por ello que el trasladar los puntos x' s, y' s del escenario Frontal se emplearán las siguientes ecuaciones.

$$x = \frac{h_{0,0}Q_x + h_{0,1}Q_y + h_{0,2}\mathcal{T}}{h_{2,0}Q_x + h_{2,1}Q_y + h_{2,2}\mathcal{T}} \quad (3.7)$$

$$y = \frac{h_{1,0}Q_x + h_{1,1}Q_y + h_{1,2}\mathcal{T}}{h_{2,0}Q_x + h_{2,1}Q_y + h_{2,2}\mathcal{T}} \quad (3.8)$$

Para las regiones de interés localizadas dentro del Centro de Proyección, se genera una matriz de medidas válidas. De esta manera cualquier punto encontrado en el plano B será trasladado al vector de medidas válidas del plano A. Aclárese que para este trabajo el Plano A, será el escenario Corredor y el plano B será el escenario Frontal.

3.5. Filtro Kalman

Para el proceso de estimación de los estados de cada trayectoria se tiene el algoritmo Kalman Lineal ó Estándar, las pruebas realizadas han sido hechas con funciones lineales con el objetivo de evaluar el correcto funcionamiento de este algoritmo, ya que posteriormente este filtro será empleado en los algoritmos de asociación.

Para empezar las matrices empleadas en este algoritmo son las siguientes:

El vector de estados x :

$$\hat{x} = \begin{pmatrix} 0,55 & 0,00022 \\ 0,55 & 0,55 \end{pmatrix} \quad (3.9)$$

La matriz de transición F , donde $T = 1,0$

$$F = \begin{pmatrix} 1,0 & 0,0 & T & 0,0 \\ 0,0 & 1,0 & 0,0 & T \\ 0,0 & 0,0 & 1,0 & 0,0 \\ 0,0 & 0,0 & 0,0 & 1,0 \end{pmatrix} \quad (3.10)$$

La matriz de covarianza H

$$H = \begin{pmatrix} 1,0 & 0,0 & 0,0 & 0,0 \\ 0,0 & 1,0 & 0,0 & 0,0 \end{pmatrix} \quad (3.11)$$

La matriz del error de covarianza Q , donde $T = 1,0$ y $s = 1,1$

$$Q = \begin{pmatrix} \frac{T^4 s}{4} & 0,0 & \frac{T^3 s}{2} & 0,0 \\ 0,0 & \frac{T^4 s}{4} & 0,0 & \frac{T^3 s}{2} \\ \frac{T^3 s}{2} & 0,0 & T^2 s & 0,0 \\ 0,0 & \frac{T^3 s}{2} & 0,0 & T^2 s \end{pmatrix} \quad (3.12)$$

La matriz P

$$P = \begin{pmatrix} 1,0 & 0,0 & 0,0 & 0,0 \\ 0,0 & 1,0 & 0,0 & 0,0 \\ 0,0 & 0,0 & 1,0 & 0,0 \\ 0,0 & 0,0 & 0,0 & 1,0 \end{pmatrix} \quad (3.13)$$

La matriz Identidad I

$$I = \begin{pmatrix} 1,0 & 0,0 & 0,0 & 0,0 \\ 0,0 & 1,0 & 0,0 & 0,0 \\ 0,0 & 0,0 & 1,0 & 0,0 \\ 0,0 & 0,0 & 0,0 & 1,0 \end{pmatrix} \quad (3.14)$$

Con el modelo de predicción se logró estimar el estado siguiente de las trayectorias generadas en cada secuencia.

3.6. NNKF

Este algoritmo funciona como el algoritmo Nearest Neighbor, la diferencia radica en que se agregó el filtro de Kalman, para realizar el proceso de estimación de las probabilidades para calcular cual de los valores tiene menor error de probabilidad. El mantenimiento se realiza de la siguiente manera.

- *Se puede asociar una medida generada a la trayectoria*
- *Existe la posibilidad de que existan más medidas, solo una pertenece a la trayectoria.*

- Cuando no se detectan medidas asociadas a la trayectoria, se estima el valor con el Filtro Kalman con 50 únicas iteraciones más, por ejemplo. Si se excede de tiempo y no hay más medidas se elimina la trayectoria.

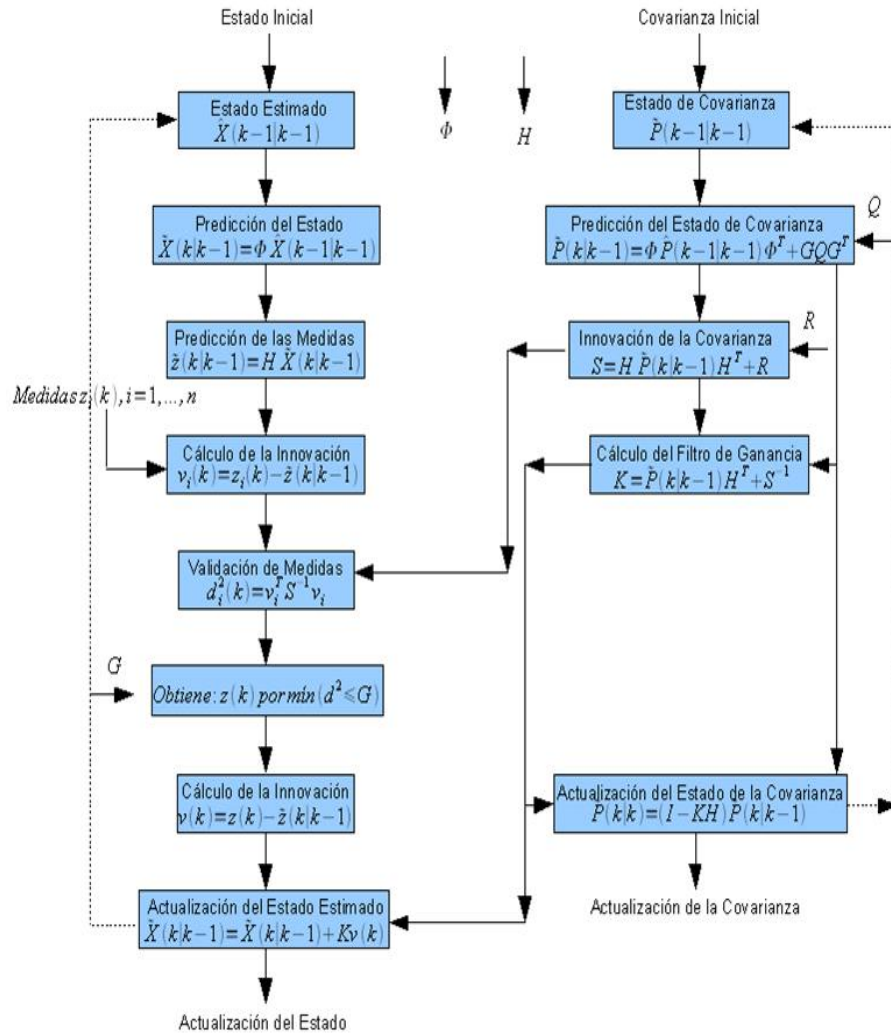


Figura 3.8: Diagrama de Flujo del Algoritmo NNKF

3.7. PDA y JPDA

PDA es un algoritmo que se utiliza para asociar datos a una trayectoria existente en este caso generada por personas, estos datos previamente se evaluaron en el umbral del centro de proyección de cada plano, luego se toman como medidas nuevas que pueden

caer dentro del área de asociación de alguna trayectoria o bien pueden ser solo ruido o algo que no pertenezca a la trayectoria. El filtro de predicción puede ayudar a eliminar el ruido existente en el estado de la trayectoria y a la vez a estimar el valor del siguiente estado, para después poder ser asociado.

En este algoritmo se deben de tener estas opciones.

- *Se puede asociar una medida generada a la trayectoria*
- *Existe la posibilidad de que existan más medidas, solo una pertenece a la trayectoria.*
- *Cuando no se detectan medidas asociadas a la trayectoria, se estima el valor con el Filtro Kalman con 55 únicas iteraciones más, por ejemplo. Si se excede de tiempo y no hay más medidas se elimina la trayectoria.*

JPDA asocia todas las medidas encontradas a las distintas trayectorias generadas en el momento actual, cabe mencionar que todos los cálculos de estimación se realizan de forma separada por cada trayectoria, no así la manera conjunta de la asociación entre la trayectoria conocida y las medidas nuevas. Una vez terminada la evaluación de todas las trayectorias se deben de tener las siguientes opciones:

- *Se puede asociar una o más medidas generadas a una de las trayectorias*
- *Cuando no se detectan medidas asociadas a alguna trayectoria, se estima el valor con el Filtro Kalman 10 únicas iteraciones más. Si se excede de tiempo y no hay más medidas se elimina la trayectoria.*
- *El vector de medidas una vez que asocia una medida elimina esa medida y trabaja con las restantes, para las siguientes trayectorias.*
- *Si existe una medida nueva y no se ha asociado a alguna trayectoria en más de 10 iteraciones se crea una trayectoria nueva.*

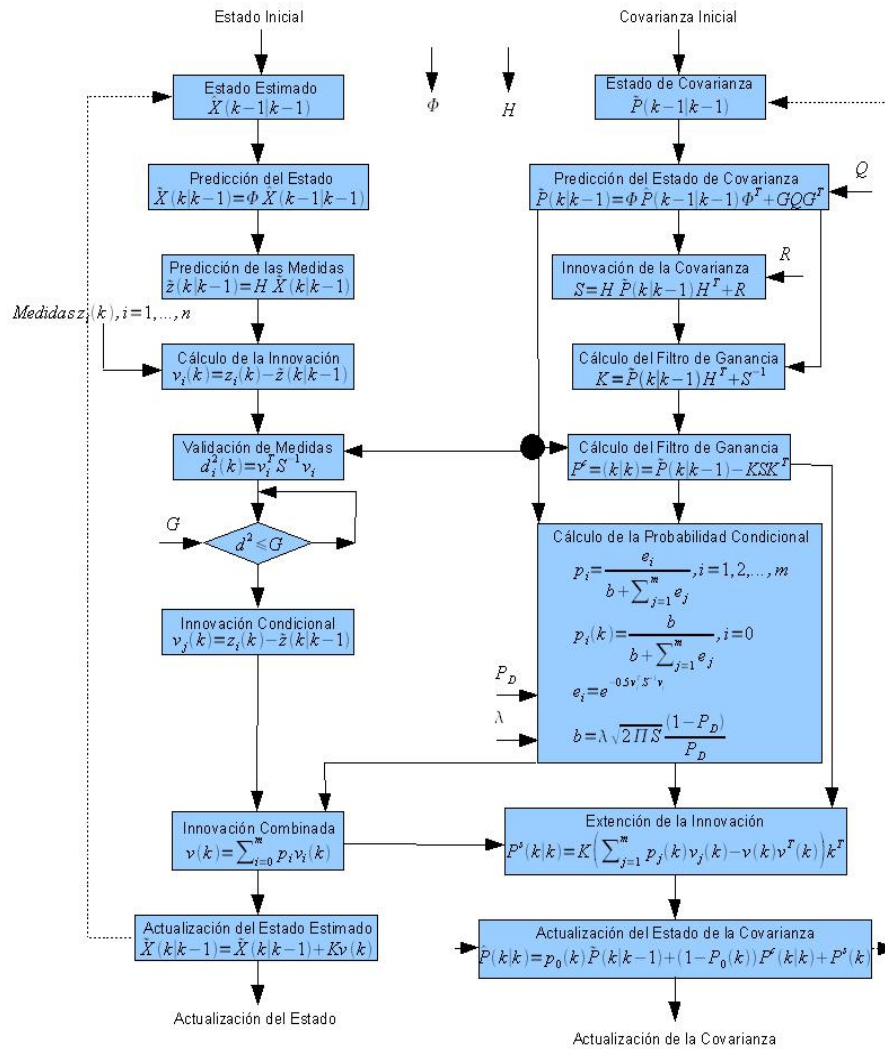


Figura 3.9: Diagrama de Flujo del Algoritmo PDA

Capítulo 4

Pruebas del Sistema de Seguimiento y Resultados

La razón por muy lógica que parezca esta conectada intimamente a por lo menos un sentimiento... Jernets

4.1. Introducción

En este capítulo se hacen las pruebas correspondientes al sistema de seguimiento de personas. Primero se realizarán pruebas para el algoritmo de segmentación, después para el proceso de la transformación proyectiva; una vez vistos los primeros procesos se harán pruebas con el algoritmo de predicción de Kalman, para continuar con los algoritmos de asociación de datos el NNKF y PDA, estos tienen en común que solo sirven para dar seguimiento a un solo objeto por lo tanto, después de hacer pruebas con cada uno de ellos se harán comparaciones entre estos dos algoritmos y finalmente se harán pruebas con el algoritmo JPDA.

4.2. Segmentación

En cada secuencia de imágenes existe un error de desfase, por lo tanto se ha hecho corresponder cada marco con una operación de sustracción entre el número total de marcos de la secuencia Corredor y la Frontal, para leer las secuencias al mismo tiempo. Por lo tanto se tienen sincronizadas las dos secuencias marco a marco.

La segmentación detecta los objetos en movimiento, sin embargo el contraste y brillo de cada escenario hacen más difícil el proceso de detección, por tal motivo se realizan diferentes operaciones a las imágenes para poder detectar las diferentes regiones de interés y coadyuvar a obtener una mayor nitidez en la imagen. Como se vio en el capítulo 3 uno de los algoritmos a emplear fue el Filtro Sobel, el cual permitió encontrar los límites de las regiones que se encuentran en movimiento.

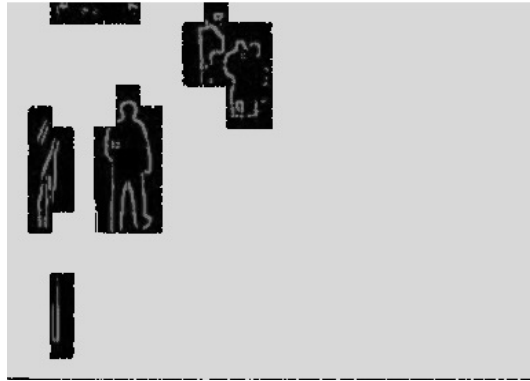


Figura 4.1: Escenario Corredor con los bordes detectados de las regiones

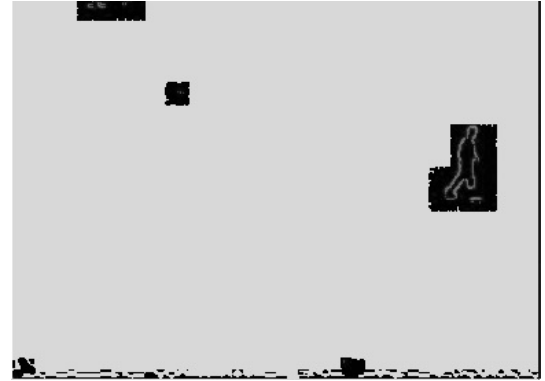


Figura 4.2: Escenario Frontal con los bordes detectados de las regiones

Una vez detectados los bordes se procede a realizar una sustracción entre la imagen base y la secuencia de imágenes, después se realiza una umbralización de las regiones con el objetivo de obtener mayor información de la region. Cabe mencionar que otro de los problemas de la detección fué la oclusión existente entre las personas, problema que no se resolvió en esta tesis.

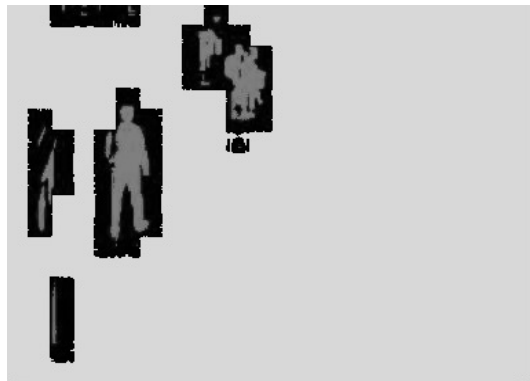


Figura 4.3: Escenario Corredor con la umbralización. Previamente se realizó una sustracción



Figura 4.4: Escenario Frontal con la umbralización. Previamente se realizó una sustracción

Una vez que se tienen las regiones detectadas en cada uno de los escenarios se combina la información y la región resultante se le encierra en una región rectangular, para poder obtener 3 puntos básicos que mantienen el área de interés. El resultado se muestra en las imágenes siguientes donde los puntos elegidos son aquellos que se encuentran más cerca del plano piso evaluansé aquellos que caen más cerca del centro de proyección.

Por lo tanto los puntos que se encuentren dentro de las coordenadas dadas en las tablas 3.1 y 3.2 serán elegidos para obtener medidas asociadas a una trayectoria existente o medidas nuevas generadas por una nueva trayectoria.



Tabla 4.1: Escenario Corredor con el objeto detectado



Tabla 4.2: Escenario Frontal con el objeto detectado

4.3. Transformación Proyectiva

Ya que se han detectado los puntos se procede a realizar la transformación proyectiva, donde uno de los sensores denominado escenario Corredor se toma como el plano base para generar la matriz homográfica y el otro sensor denominado escenario Frontal, transforma los puntos localizados en su plano base de acuerdo a los valores generados por la matriz homográfica.

Los puntos convergentes de cada escenario del proyecto CAVIAR, están dadas en el capítulo anterior, por lo tanto en este capítulo se harán las pruebas para obtener los valores generados en la matriz homográfica.

$$H_x = \begin{bmatrix} 0,000780983 & 0,000378913 & 0,0179301 \\ -0,00333826 & -8,97563e^{005} & 0,999833 \\ 5,27707e^{-006} & 6,01346e^{008} & 0,000308087 \end{bmatrix} \quad (4.1)$$



Figura 4.5: Transformación del escenario Frontal, al plano Corredor



Figura 4.6: Puntos localizados dentro del Centro de Proyección, en el escenario Corredor



Figura 4.7: Puntos localizados dentro del Centro de Proyección, en el escenario Frontal

4.4. Pruebas con el Filtro Kalman Lineal

En esta sección se realizarón la primeras pruebas con ecuaciones lineales para conocer los resultados del filtro Kalman, después se realizaron algunas pruebas con las secuencias de imágenes utilizando ya el algoritmo de detección, cabe mencionar que las ecuaciones son estrictamente lineales, ya que no se esta agregando la dirección del objeto o la aceleración, el sistema solo obtiene la posición (x, y) de cada punto, este problema lo resuelve el Filtro Kalman Lineal, dada la naturaleza de las trayectorias.

4.4.1. Ecuación Lineal con el Filtro Kalman

Una de las funciones empleadas fue $f(x) = x$, donde el valor de x esta dado por el número de la iteración, por lo tanto la ecuación es creciente, donde el filtro Kalman estimó los puntos generados y después se procedió a graficar cada punto obtenido.

En la tabla y gráfica siguientes se muestran algunos de los valores obtenidos, además las celdas de color magenta marcan en que momento el punto (x', y') logra aproximarse al valor original del punto (x, y) , respectivamente.

Punto	(x, y)	(x', y')
1	(1, 1)	(1.55, 1.55)
2	(2, 2)	(2.474, 2.340)
3	(3, 3)	(3.767, 3.436)
⋮	⋮	⋮
68	(68, 68)	(68.998, 68.999)
69	(69, 69)	(69.999, 69.999)
70	(70, 70)	(71.000, 70.999)
⋮	⋮	⋮
115	(115, 115)	(116.000, 116)
116	(116, 116)	(117.000, 117)
117	(117, 117)	(118, 118)
⋮	⋮	⋮
302	(302, 302)	(303, 303)
303	(303, 303)	(304, 304)
304	(304, 304)	(305, 305)

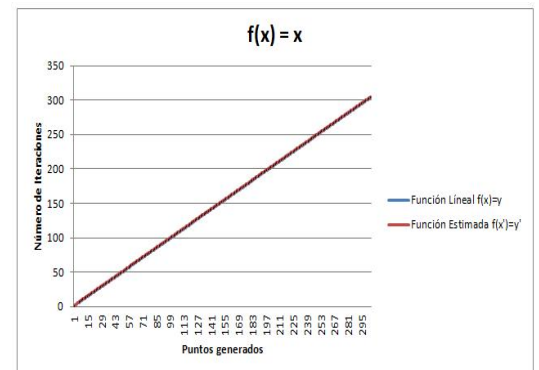


Tabla 4.3: Puntos de la función $f(x) = x$

En este caso el Filtro Kalman Lineal cumple con la estimación de manera correcta dado que la ecuación original se ha estimado sin agregarse algún tipo de ruido. Es importante mencionar que este filtro necesita tener cierto número de iteraciones para poder adaptarse a la trayectoria original y así efectuar la estimación de los puntos de

manera ideal, sin embargo esto no es fácil de obtener si hay mucha variabilidad en el cálculo de los puntos.

Para una segunda prueba la ecuación empleada fue $f(x) = 3x$, donde los puntos generados se muestran en la tabla y gráfica siguiente:

Punto	(x, y)	(x', y')
1	(1, 3)	(1.55, 3.55)
2	(2, 6)	(2.474, 5.409)
3	(3, 9)	(3.767, 8.930)
4	(4, 12)	(5.157, 13.243)
⋮	⋮	⋮
80	(80, 240)	(81.000, 242.999)
81	(81, 243)	(81.999, 245.999)
82	(82, 246)	(82.999, 249)
83	(83, 249)	(83.999, 252)
⋮	⋮	⋮
302	(302, 906)	(303, 909)
303	(303, 909)	(304, 912)
304	(304, 912)	(305, 915)

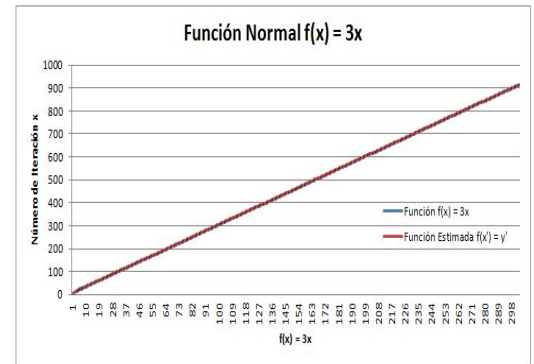


Tabla 4.4: Tabla y gráfica de la ecuación lineal $f(x) = 3x$ y $f(x') = 3x'$

De igual manera esta ecuación no se vió afectada por algún tipo de ruido y el algoritmo de estimación efectuó su función de manera correcta.

A las siguientes pruebas se les agregó ruido en su trayectoria original causando falsas detecciones así como cambios repentinos en los puntos originales, con el fin de comprobar que tan robusto es el Filtro Kalman Lineal, como se mencionó anteriormente este algoritmo funciona cuando las trayectorias no cambian tan drásticamente en las iteraciones.

Los resultados obtenidos se muestran en la tabla siguiente, el ruido sí afecta a la trayectoria original, pero si hay una secuencia frecuente de puntos lineal la estimación se va acercando a la secuencia de puntos de la trayectoria original, por lo tanto no hay tanto problema en la estimación de los puntos de la trayectoria original.

Punto	(x, y)	(x', y')
1	(1, 3)	(1.55, 3.55)
2	(2, 6)	(2.474, 5.409)
3	(3, 9)	(3.767, 8.930)
⋮	⋮	⋮
48	(48, 144)	(49.003, 146.999)
49	(49, 147)	(50.006, 149.999)
50	(50, 150)	(51.004, 153.000)
51	(51, 51)	(52.000, 93.081)
⋮	⋮	⋮
75	(75, 75)	(75.999, 75.469)
76	(76, 76)	(76.999747, 76.328)
77	(77, 77)	(78.000, 77.427)
78	(78, 78)	(79.000, 78.660)
⋮	⋮	⋮
302	(302, 906)	(303, 905.817)
303	(303, 909)	(304, 912.669)
304	(304, 912)	(305, 918.069)

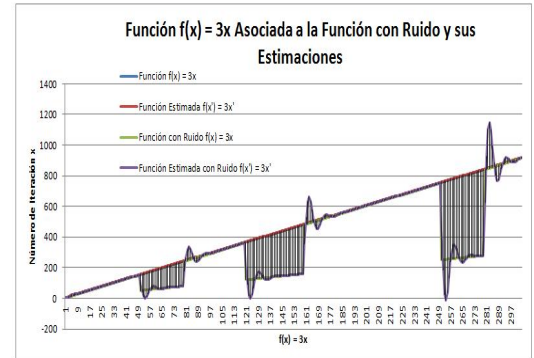
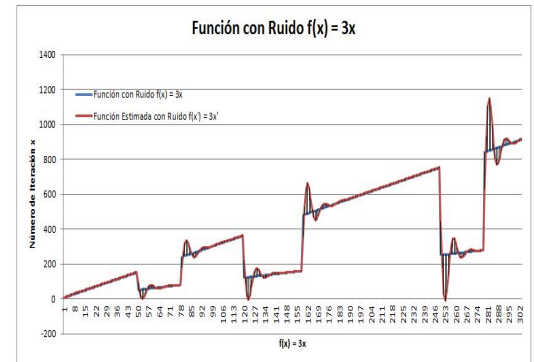


Tabla 4.5: Tabla y gráfica de la ecuación lineal $f(x) = 3x$ y $f(x') = 3x'$ con ruido. En la gráfica siguiente se muestran las diferencias en las funciones originales y estimaciones, con errores. Como se puede apreciar el cálculo de la estimación se vió afectado por el ruido existente generado, sin embargo el algoritmo logró reponerse a los cambios y prosiguió estimando los puntos de manera correcta.

4.4.2. Secuencias de Imágenes con el Filtro Kalman

Por ejemplo, en las pruebas siguientes el Filtro Kalman Lineal obtuvo la predicción para los puntos (x, y) y (x', y') de las secuencias de imágenes, sin embargo hay secciones de la gráfica donde se ve afectada la estimación, debido al cambio repentino de los puntos de la secuencia original, estos cambio repentinos se deben a que el algoritmo de seguimiento no detecto de manera correcta los puntos cercanos al punto anterior o el ruido afecto en el cálculo de la estimación.

Frame	(x, y) Real	(x' , y') Estimado
0	(163, 298.955)	(0, 0)
2	(163, 298.955)	(163.55, 299.505)
3	(163, 298.955)	(163.642, 299.761)
4	(165, 293)	(164.925, 295.997)
⋮	⋮	⋮
184	(169, 300.857)	(165.238, 287.855)
185	(167, 298.933)	(162.732, 299.986)
186	(166, 299.296)	(162.581, 308.074)
187	(165, 293)	(163.840, 307.089)
⋮	⋮	⋮
301	(175, 300.619)	(174.504, 269.079)
302	(175, 300.638)	(176.382, 289.226)
303	(175, 300.666)	(177.052, 306.125)
304	(175, 300.666)	(176.290, 315.921)

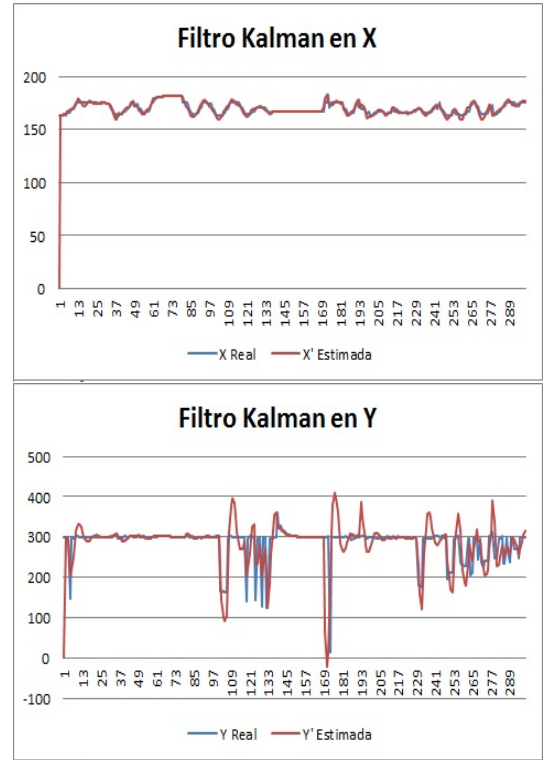


Tabla 4.6: Tabla y gráfica de la secuencia de imagen 1.

Esta prueba proporciona un margen de error pequeño de los puntos detectados respecto a los estimados, cabe mencionar que cada estimación se ha visto afectada por la existencia de falsas detecciones o ruido.

Una segunda prueba muestra los valores obtenidos por la estimación del Filtro Kalman Lineal y los puntos detectados en la secuencia original.

Las pruebas realizadas en el algoritmo de estimación se hicieron con ecuaciones lineales y con secuencias de imágenes. Cada prueba realizada a las funciones lineales se les agregaron ecuaciones que causaron ruido a la trayectoria original y las realizadas con las secuencias de imágenes fueron afectadas de algún modo por las detecciones falsas o el ruido. De esta manera se logró comprobar lo robusto que es el Filtro Kalman Lineal.

Frame	(x, y) Real	(x', y') Estimado
0	(166, 300.039)	(166.55, 300.589)
1	(167, 295.129)	(167.474, 298.221)
2	(167, 294.738)	(167.946, 294.917)
⋮	⋮	⋮
131	(166, 297.030)	(167.228, 296.751)
132	(166, 301.5)	(166.982, 298.531)
133	(166, 301.5)	(166.146, 300.593)
⋮	⋮	⋮
178	(163, 248.727)	(161.544, 235.189)
179	(163, 179.5)	(164.145, 186.386)
180	(163, 180)	(165.568, 151.828)

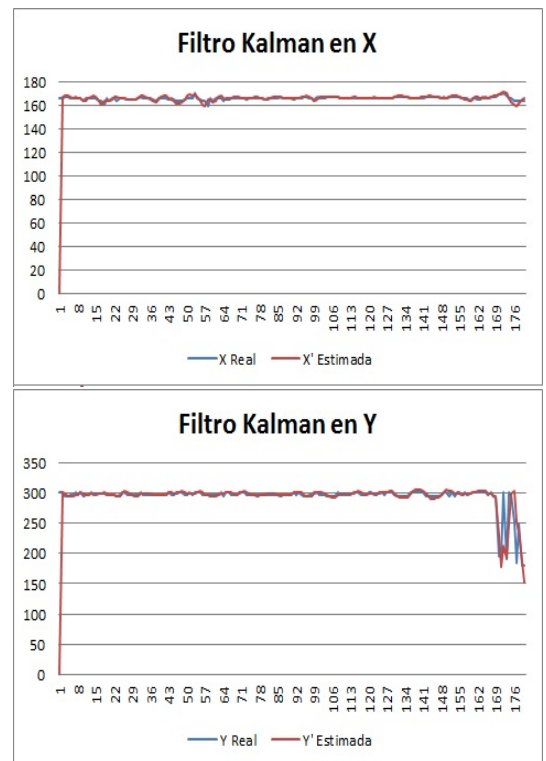


Tabla 4.7: Tabla y gráfica de secuencia de imagen 2.

4.5. Pruebas con el algoritmo NNKF

Las pruebas realizadas en esta sección pertenecen a los escenarios del proyecto CAVIAR.

OneLeaveShop1:

Frame	(x, y) Real	(x', y') Estimado
95	(180, 95)	(180, 95.000)
96	(180, 95)	(180.55, 95.550)
97	(180, 98)	(180.648, 97.600)
98	(180, 91)	(180.316, 94.817)
⋮	⋮	⋮
112	(176, 111)	(175.902, 112.454)
113	(177, 105)	(176.834, 109.305)
114	(177, 105)	(177.714, 105.934)
115	(176, 105)	(177.173, 103.393)
⋮	⋮	⋮
124	(172, 110)	(168.792, 111.546)
125	(169, 112)	(172.592, 110.742)
126	(174, 116)	(171.802, 114.469)
127	(174, 117)	(172.558, 117.894)

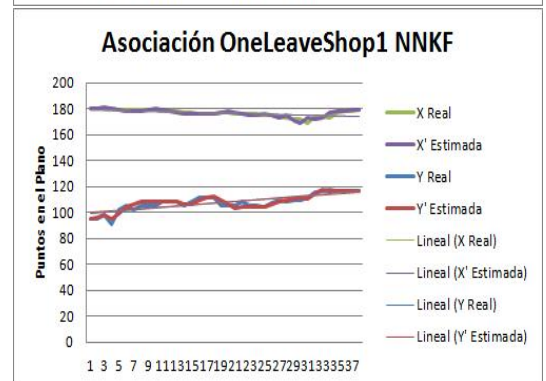


Tabla 4.8: Tabla y gráficas de los puntos asociados del escenario OneLeaveShop1 con el algoritmo NNKF.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.870481772 y para los puntos (y, y') es 0.952526294, lo cual son niveles altos de correlación, por lo tanto se entiende que las trayectorias en ambos pares de puntos (x, x') y (y, y') están correlacionadas fuertemente entre sí. La Línea de Tendencia Li-

neal muestra claramente el sentido descendente de los puntos en el eje x y en el eje y estan de manera ascendente.

Las pruebas siguientes fueron realizadas con el escenario llamado OneLeaveShop2:

Frame	(x, y) Real	(x', y') Estimado
1276	(199, 93)	(199.55, 93.55)
1277	(199, 199)	(199.642, 150.452)
1278	(199, 93)	(199.284, 149.054)
1279	(199, 199)	(198.807, 196.214)
⋮	⋮	⋮
1319	(202, 126.5)	(204.600, 132.860)
1320	(205 112)	(208.084, 123.749)
1321	(208 113)	(211.432, 114.818)
1322	(208 120)	(212.005, 112.748)
⋮	⋮	⋮
1434	(264, 174)	(264.116, 180.877)
1435	(260, 166)	(262.474, 176.471)
1436	(263, 166)	(261.958, 170.608)
1437	(262, 171)	(262.583, 176.234)



Tabla 4.9: Tabla y gráficas de los puntos asociados del escenario OneLeaveShop2 con el algoritmo NNKF.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.973137635 y para los puntos (y, y') es 0.963886744. La Línea de Tendencia Lineal muestra claramente el sentido descendente de las puntos en (x, x') , y ascendente en los puntos (y, y') , sin embargo la correlación muestra que las trayectorias en ambos pares de puntos (x, x') y (y, y') estan fuertemente correlacionadas entre si.

4.6. Pruebas con el algoritmo PDA

Las pruebas realizadas en esta sección pertenecen a los escenarios del proyecto CAVIAR.

OneLeaveShop1:

Frame	(x, y) Real	(x', y') Estimado
95	(180, 95)	(180, 95)
96	(180, 95)	(18.55, 96)
97	(180, 98)	(180.776, 97.176)
98	(177, 96)	(180.487, 95.661)
⋮	⋮	⋮
112	(176, 111)	(174.464, 110.497)
113	(182, 95)	(175.418, 109.802)
114	(182, 95)	(176.988, 108.048)
115	(186, 99)	(177.831, 105.976)
⋮	⋮	⋮
124	(172, 110)	(169.318, 110.207)
125	(169, 110)	(172.392, 111.042)
126	(174, 116)	(171.197, 113.533)
127	(174, 117)	(170.706, 115.816)

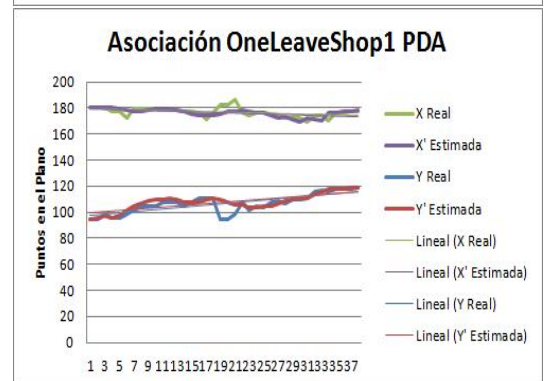
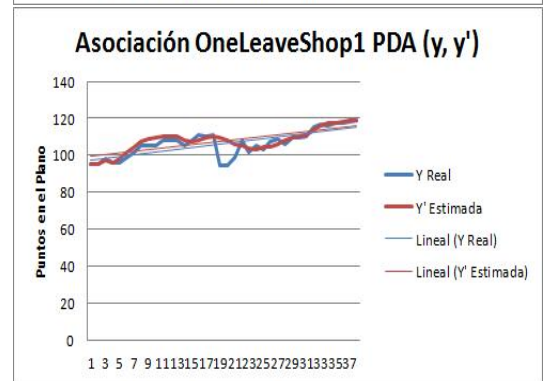


Tabla 4.10: Tabla y gráficas de los puntos asociados del escenario OneLeaveShop1 con el algoritmo PDA.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.614003396 y para los puntos (y, y') es 0.852462698, en el primer caso se entiende que la tendencia correlacional no es tan fuerte, sin embargo se mantiene en un buen rango por arriba del .5, la Línea de Tendencia Lineal muestra claramente

el sentido descendente en los puntos (x, x') sin embargo en los puntos (y, y') esto es inverso.

Una vez realizadas las pruebas con las funciones lineales se procede a generar las pruebas con el escenario OneLeaveShop2:

Frame	(x, y) Real	(x', y') Estimado
1276	(199, 93)	(199.55, 93.55)
1277	(199, 199)	(199.765, 135.231)
1278	(199, 93)	(199.593, 141.756)
1279	(199, 199)	(199.112, 159.667)
⋮	⋮	⋮
1319	(203, 117)	(198.037, 95.629)
1320	(207, 119)	(198.380, 97.924)
1321	(207, 119)	(199.225, 104.004)
1322	(206, 121)	(200.669, 111.190)
⋮	⋮	⋮
1423	(257, 174)	(267.162, 155.595)
1424	(256, 166)	(268.466, 157.453)
1425	(243, 168)	(268.946, 160.460)
1426	(265, 171)	(268.914, 163.075)

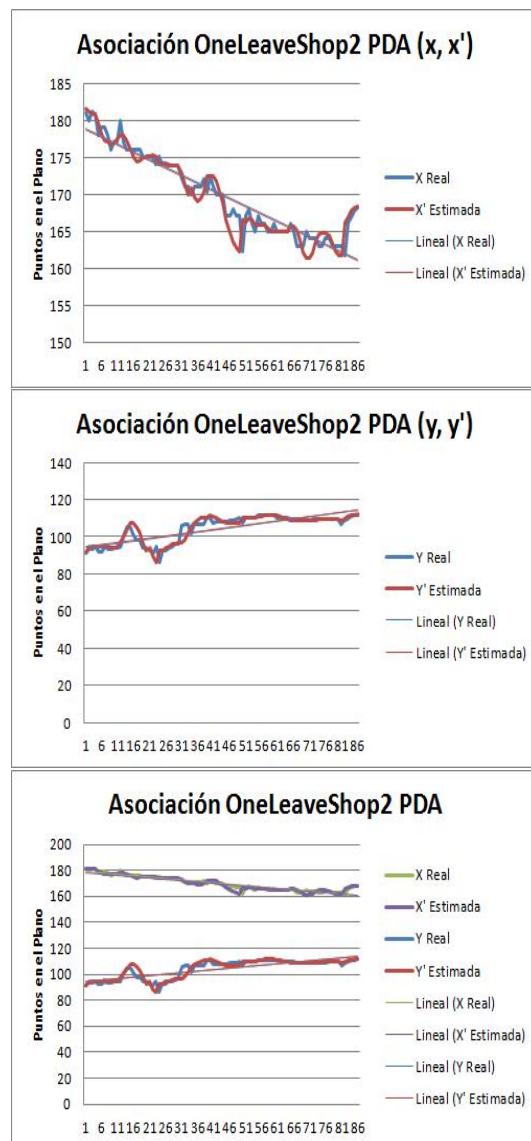


Tabla 4.11: Tabla y gráficas de los puntos asociados del escenario OneLeaveShop2 con el algoritmo PDA.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.96357246 y para los puntos (y, y') es 0.936172971, en este último caso se entiende que la tendencia correlacional es fuerte. La Linea de Tendencia Lineal muestra claramente el sentido descendente en los puntos (x, x') sin embargo en los puntos (y, y') es un caso inverso.

4.7. Pruebas con el algoritmo JPDA

Las pruebas realizadas con el algoritmo JPDA fueron realizadas al varios escenarios entre ellos el escenario llamado EnterExitCrossingPaths1, para estas pruebas se busco que entre las personas encontradas dentro del centro de proyeccion no se estorbaran, con el objetivo de evitar la oclusión, ya que el algoritmo de seguimiento no resuelve este problema. La principal diferencia entre JPDA y los dos primeros algoritmos NNKF y PDA, es que este algoritmo en su proceso de mantenimiento crea más trayectorias nuevas para dar seguimiento a los objetos nuevos que aparezcan en el escenario.

A continuación se muestran los resultados obtenidos con este algoritmo con el escenario EnterExitCrossingPaths1 con el objeto 1.

Frame	(x, y) Real	(x', y') Estimado
98	(163, 164)	(163, 164)
99	(163, 167)	(163.55, 165.538)
100	(163, 165)	(163.776, 166.215)
101	(163, 163)	(163.624, 165.713)
⋮	⋮	⋮
126	(172, 138)	(170.686, 142.361)
127	(172, 123)	(171.100, 135.495)
128	(168, 126)	(171.465, 127.201)
129	(172, 120)	(172.122, 118.681)
⋮	⋮	⋮
153	(181, 95)	(182.473, 86.162)
154	(181, 95)	(181.969, 90.573)
155	(180, 95)	(180.718, 94.675)
156	(180, 95)	(179.241, 98.099)

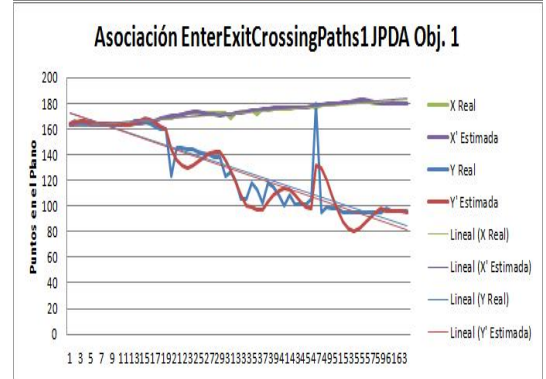
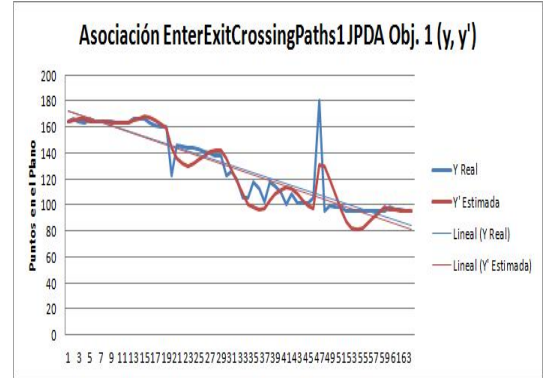
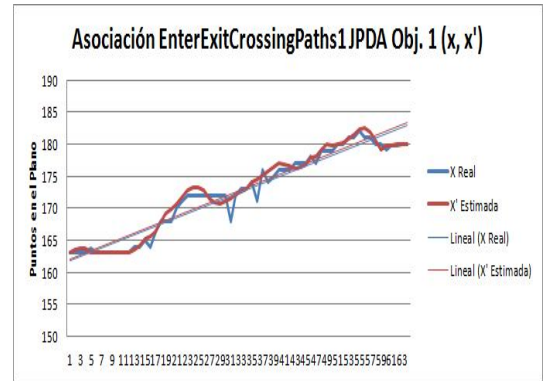


Tabla 4.12: Tabla y gráficas de los puntos asociados a la trayectoria del objeto 1 del escenario EnterExitCrossingPaths1 con el algoritmo JPDA.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.990500276 lo cual es un coeficiente correlacional muy fuerte y para los puntos (y, y') es 0.932310889, en este último caso se entiende que el coeficiente correlacional también es fuerte. Las Líneas de Tendencia Lineal muestran claramente el sentido ascendente en los puntos (x, x') y descendente en los puntos (y, y') .

Los resultados obtenidos en las siguientes pruebas fueron realizados en el mismo escenario con el objeto 2.

Frame	(x, y) Real	(x', y') Estimado
212	(208, 175)	(208, 174.499)
213	(208, 176)	(208.549, 175.445)
214	(209, 177)	(209.366, 176.693)
215	(210, 170)	(210.494, 174.525)
⋮	⋮	⋮
251	(229, 208)	(226.489, 205.607)
252	(225, 200)	(225.741, 203.375)
253	(225, 203)	(224.806, 202.118)
254	(225, 204)	(224.057, 201.746)
⋮	⋮	⋮
286	(253, 248)	(252.227, 235.343)
287	(251, 231)	(255.692, 233.370)
288	(254, 256)	(259.696, 236.286)
289	(172, 249)	(264.010, 236.848)

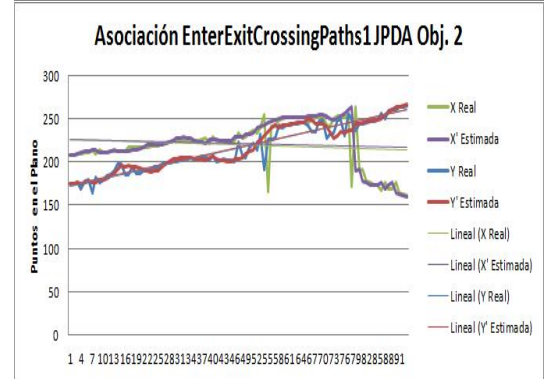
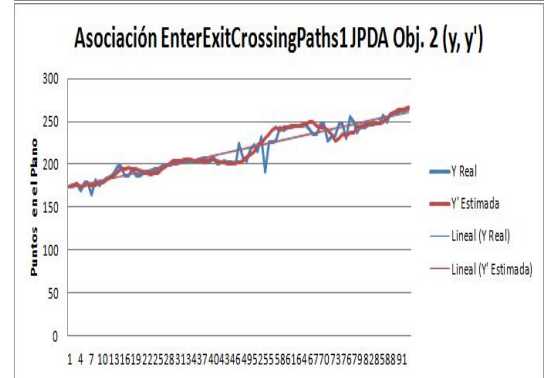
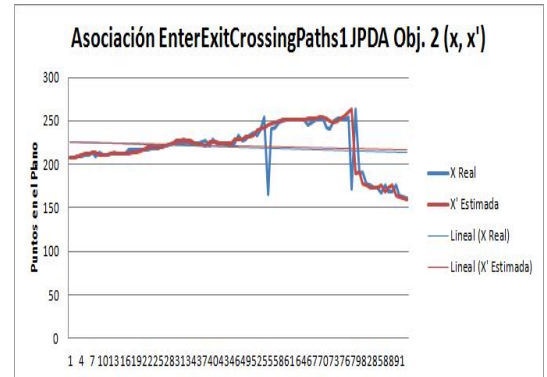


Tabla 4.13: Tabla y gráficas de los puntos asociados al objeto 2 del escenario EnterExit-CrossingPaths1 con el algoritmo JPDA.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.826900085 lo cual es un coeficiente correlacional muy fuerte y para los puntos (y, y') es 0.962206228, en este último el coeficiente correlacional también es fuerte. Las Líneas de Tendencia Lineal muestran el sentido ascendente en los puntos, se tiene que la trayectoria tiende a ser ascendente.

Las pruebas realizadas a continuación fueron hechas con el escenario EnterExit-CrossingPaths2, en las siguientes tablas y gráficas se muestran los resultados.

Frame	(x, y) Real	(x', y') Estimado
221	(164, 108)	(164.55, 108.3)
222	(165, 108)	(164.642, 108.556)
223	(165, 109)	(164.735, 108.775)
224	(163, 106)	(164.428, 108.283)
⋮	⋮	⋮
333	(170, 125)	(171.150, 126.543)
334	(171, 125)	(171.985, 126.698)
335	(171, 125)	(171.705, 125.798)
336	(170, 125)	(170.261, 126.610)
⋮	⋮	⋮
444	(188, 141)	(187.051, 133.589)
445	(188, 141)	(186.074, 138.341)
446	(198, 128)	(185.506, 140.985)
447	(198, 128)	(185.060, 143.668)

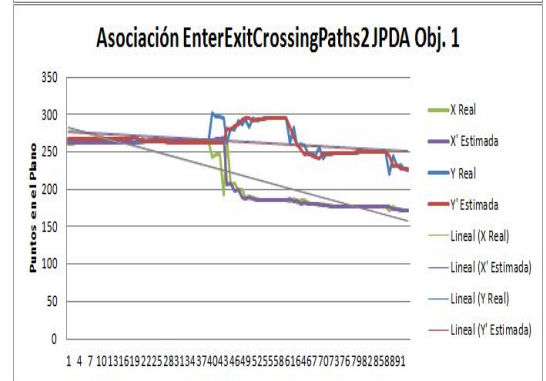
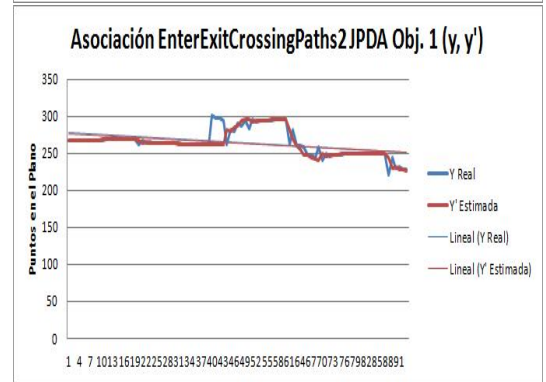
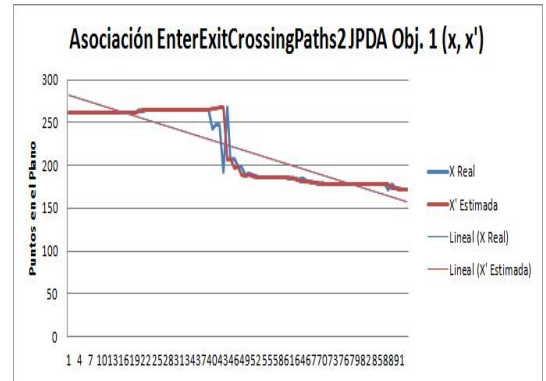


Tabla 4.14: Tabla y gráficas de los puntos asociados a la trayectoria del objeto 1 del escenario OneStopNoEnter2 con el algoritmo JPDA.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.963270572 lo cual es un coeficiente correlacional muy fuerte y para los puntos (y, y') es 0.897622605, en este último caso se entiende que el coeficiente correlacional tiende a ser fuerte. Las Líneas de Tendencia Lineal muestran el sentido descendente en los puntos (x, x') y (y, y') , por lo tanto, se tiene que la trayectoria tiende a ser descendente.

Pruebas en EnterExitCrossingPaths2 con un segundo objeto detectado.

Frame	(x, y) Real	(x', y') Estimado
579	(227, 147)	(227, 146.999)
580	(227, 148)	(227.320, 145.393)
581	(227, 149)	(227.331, 142.495)
582	(227, 150)	(227.354, 139.784)
⋮	⋮	⋮
596	(227, 148)	(227.625, 141.117)
597	(227, 147)	(228.794, 138.349)
598	(227, 146)	(228.486, 140.263)
599	(227, 146)	(227.469, 142.833)
⋮	⋮	⋮
612	(240, 137)	(242.230, 138.310)
613	(240, 137)	(244.065, 134.818)
614	(251, 132)	(247.737, 122.365)
615	(240, 145)	(251.951, 114.265)

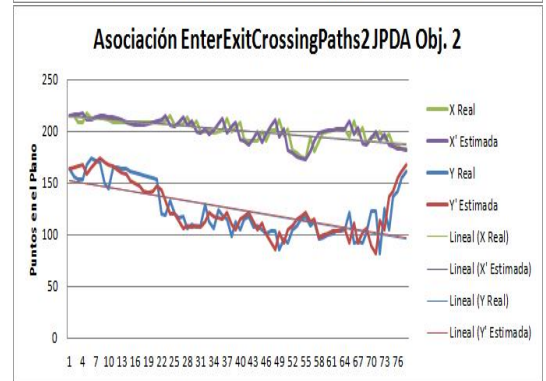
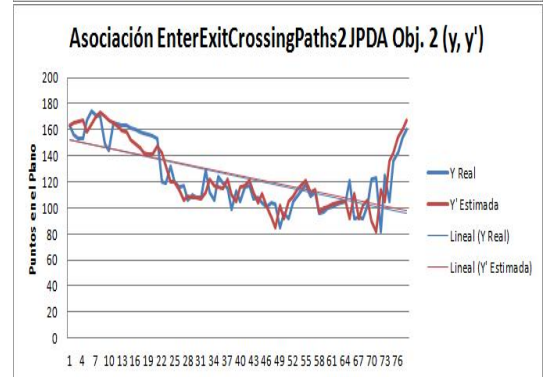
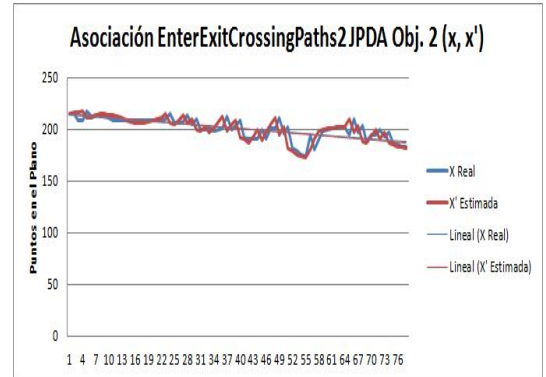


Tabla 4.15: Tabla y gráficas de los puntos asociados a la trayectoria del objeto 2 del escenario EnterExitCrossingPaths2 con el algoritmo JPDA.

El valor obtenido entre el Coeficiente Correlacional Lineal para los puntos (x, x') es igual a 0.801649288 lo cual es un coeficiente correlacional fuerte y para los puntos (y, y') es 0.87597552, en este último caso igual el coeficiente correlacional tiende a ser fuerte. Las Líneas de Tendencia Lineal muestran el sentido descendente en los puntos (x, x') y (y, y') , por lo tanto, se tiene que la trayectoria tiende a ser descendente y la correlación entre los puntos existe.

Capítulo 5

Resultados

Nuestra recompensa se encuentra en el esfuerzo y no en el resultado. Un esfuerzo total es una victoria completa. Mahatma Gandhi

5.1. Introducción

En esta sección se muestran los resultados obtenidos en cada uno de los procesos del sistema de seguimiento de personas, además se realizan las comparaciones entre los algoritmos de asociación entre el NNKF y el PDA, esta comparación se realiza ya que ambos algoritmos son empleados para el seguimiento de un solo objeto. El algoritmo JPDA le da seguimiento a varios objetos.

5.2. Resultados en la Detección

Los escenarios mostrados en esta tesis fueron grabados en un ambiente abierto, por lo tanto el algoritmo de detección a emplear tenía que soportar al máximo las problemáticas de este tipo de ambientes. Se realizaron pruebas con objetos que fueran fácilmente detectables y que en el transcurso de las secuencias no se perdiera demasiado el objeto. Por lo tanto se tuvieron que eliminar secuencias de escenarios donde la oclusión generaría un problema. Además el contraste de algunos de los objetos respecto a la imagen de fondo y la falta de homogeneidad afecto de igual manera en la detección, como resultado se tuvieron algunas falsas detecciones y se generó ruido.

Los problemas encontrados con el sensor Frontal regularmente se tuvieron a causa de el contraste respecto al fondo de la imagen ya que eran los puntos extremos al centro de proyección donde regularmente se iban perdiendo los objetos y los problemas mayormente encontrados en el escenario Corredor, fueron principalmente los relacionados a la oclusión y el contraste, ambos escenarios tuvieron problemas con las falsas detecciones y el ruido.



Figura 5.1: Escenario Corredor objeto detectado dentro del centro de proyección



Figura 5.2: Escenario Frontal con el objeto no detectado en el centro de proyección

5.3. Resultados en la Transformación Projectiva

El punto a favor en este proceso fue que el proyecto Caviar en su página proporciona los puntos correlacionados de cada escenario, obteniendo la generación de la matriz de homografía, de hecho una vez obtenidos estos puntos y generada la matriz se procedió a emplear las ecuaciones 3.7 y 3.8, para realizar el traslado de los puntos del escenario Frontal al escenario Corredor, este último tomando la función de plano base.

Los resultados fueron óptimos.

5.4. Resultados Filtro Kalman Lineal

Los resultados obtenidos en este filtro fueron evaluados en ecuaciones lineales sin y con ruido en ellas, las pruebas se mantuvieron sin mucho problema ya que el número de iteraciones fueron bastantes, esto le permitió al algoritmo llegar a estimar los valores de cada ecuación de manera correcta.

Punto	y	y'
1	0.0024	0.972669622
2	0.0032	0.67719362
3	0.004	0.267813837
4	0.0048	-0.156726235
5	0.0056	-0.504293494
⋮	⋮	⋮
77	20.328	20.336
78	20.336	20.344
79	20.344	20.352
⋮	⋮	⋮
80	20.584	20.592
81	20.592	2.06
82	2.06	20.608
⋮	⋮	⋮
302	20.776	20.784
303	20.784	20.792
304	20.792	2.08

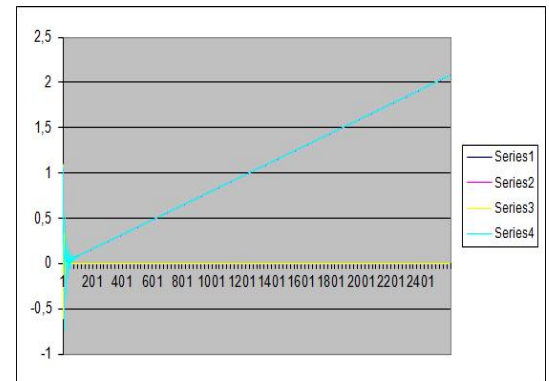


Tabla 5.1: Puntos de la función $f(x) = x$

En las pruebas con secuencias cortas de imagen los resultados fueron de igual manera favorables. Ya que estas imágenes fueron tomadas de un ambiente cerrado sin perturbaciones en la detección. Por lo tanto la estimación de los valores se realizó de manera correcta.

5.5. Resultados y Comparaciones entre NNKF y PDA

Como se mencionó anteriormente antes de realizar pruebas con los escenarios, se realizaron pruebas con ecuaciones lineales que se les agregaba ruido. Los resultados obtenidos fueron óptimos ya que se tuvo cierta ventaja de generar los puntos de cada ecuación de manera controlada y la asociación en los puntos era más fácil, las iteraciones le daban tiempo a los algoritmos asociar las medidas a la trayectoria generada.

5.5.1. Resultados NNKF

Las pruebas realizadas al algoritmo de asociación NNKF se realizaron en distintos escenarios donde solo aparece un único individuo en escena. Estas generaron resultados favorables al NNKF ya que apesar que se tuvieron problemas en la detección la asociación se estimo de manera óptima.

Los resultados se pueden apreciar en la tabla siguiente, donde se obtuvo el Coeficiente de Correlación Lineal r , para poder correlacionar las trayectorias generadas y las medidas nuevas generadas asociadas a cada trayectoria.

Algoritmo	Escenario	$r(x, x')$	%	$r((y, y'))$	%
NNKF	OneLeaveShop1	0.870481772	87.04	0.952526294	95.25
NNKF	OneLeaveShop2	0.973137635	97.31	0.963886744	96.38
NNKF	OneLeaveShopReenter2	0.995700148	99.57	0.930624592	93.06
NNKF	OneStopMoveEnter1	0.98277096	98.27	0.881147466	88.11
NNKF	OneStopMoveEnter2	0.997918509	99.79	0.99617909	99.61
NNKF	OneStopMoveNoEnter1	0.986638823	98.66	0.978745183	97.87
NNKF	OneStopNoEnter1	0.999253501	99.92	0.986187247	98.61
NNKF	TwoEnterShop1	0.998103096	99.81	0.945509652	94.55
NNKF	TwoEnterShop2	0.999714494	99.97	0.983277727	98.32
NNKF	TwoLeaveShop1	0.993080627	99.30	0.789927156	78.99

Tabla 5.2: Coeficiente de Correlación Lineal en el seguimiento empleando el algoritmo NNKF.

5.5.2. Resultados PDA

De la misma forma que en NNKF, PDA perdio en distintas iteraciones los puntos acausa de la asociacion de medidas falsas o perdida de los puntos. En ocasiones la asociacion se alejo mucho de la trayectoria original, aunque al final lograba acoplarse a esta, mejoro respecto a lo obtenido a la asociación de datos, pero no fué mejor que NNKF.

Algoritmo	Escenario	$r(x, x')$	%	$r((y, y'))$	%
PDA	OneLeaveShop1	0.614003396	61.40	0.852462698	85.24
PDA	OneLeaveShop2	0.96357246	96.35	0.936172971	93.61
PDA	OneLeaveShopReenter2	0.924639494	92.46	0.949053571	94.90
PDA	OneStopMoveEnter1	0.854751023	85.47	0.852567955	85.25
PDA	OneStopMoveEnter2	0.983739397	98.37	0.982200714	98.22
PDA	OneStopMoveNoEnter1	0.92806906	92.80	0.93595452	93.59
PDA	OneStopNoEnter1	0.998683364	99.86	0.976568166	97.65
PDA	TwoEnterShop1	0.98436486	98.43	0.948090164	94.80
PDA	TwoEnterShop2	0.9995974	99.95	0.981043357	98.10
PDA	TwoLeaveShop1	0.998106437	99.81	0.419312621	41.93

Tabla 5.3: Coeficiente de Correlación Lineal en el seguimiento empleando el algoritmo PDA.

5.6. Comparaciones NNKF y PDA

La tabla siguiente muestra el Coeficiente de Correlación de cada algoritmo de asociación, respecto a cada uno de los escenarios a los que se les hicieron pruebas.

Cabe destacar que la mayoría de los resultados están arriba del 0.5, que en sí es un porcentaje favorable, ya que esto nos asegura que la asociación se pudo realizar de manera correcta.

Escenario	Algoritmo NNKF $r(x, x')$	Algoritmo PDA $r(x, x')$	Algoritmo NNKF $r(y, y')$	Algoritmo PDA $r(y, y')$
OneLeaveShop1	0.870481772	0.614003396	0.952526294	0.852462698
OneLeaveShop2	0.973137635	0.96357246	0.963886744	0.936172971
OneLeaveShopReenter2	0.995700148	0.924639494	0.930624592	0.949053571
OneStopMoveEnter1	0.98277096	0.854751023	0.881147466	0.852567955
OneStopMoveEnter2	0.997918509	0.983739397	0.99617909	0.982200714
OneStopMoveNoEnter1	0.986638823	0.92806906	0.978745183	0.93595452
OneStopNoEnter1	0.999253501	0.998683364	0.986187247	0.976568166
TwoEnterShop1	0.998103096	0.98436486	0.945509652	0.948090164
TwoEnterShop2	0.999714494	0.9995974	0.983277727	0.981043357
TwoLeaveShop1	0.993080627	0.998106437	0.789927156	0.419312621

Como se puede apreciar el Coeficiente de Correlación más óptimo fue el obtenido por NNKF, sin embargo hubo dos escenarios donde la asociación costó un poco más para este que fue TwoLeaveShop1 donde en el eje de las $x's$, no tuvo una aproximación fuerte. Y para PDA el escenario donde tuvo más problema fue OneLeaveShopOne1, donde en puntos $x's$ y $y's$, estuvo por debajo del 0,9%.

5.7. Resultados JPDA

Al igual que los otros dos algoritmo arrojaron resultados favorables con pruebas realizadas empleando ecuaciones lineales. El algoritmo ejecutado con los escenarios tuvo los mismos problemas en la detección ya que tampoco fue favorable.

A este algoritmo se le agregó en el proceso de mantenimiento el poder agregar trayectorias generadas por medidas nuevas en tiempo de ejecución, resultado que si pudo generarlas en las pruebas con las ecuaciones lineales y con los escenarios, además se mostro muy robusto con el ruido agregado a las trayectorias.

En las pruebas realizadas a los distintos escenarios arrojaron los siguientes resultados.

Algoritmo	Escenario	$r(x, x')$ Obj 1	%	$r(y, y')$ Obj. 1	%	$r(x, x')$ Obj 2	%	$r(y, y')$ Obj. 2	%
JPDA	EnterExitCrossingPaths1	0.990500276	99.05	0.932310889	93.23	0.826900085	82.69	0.962206228	96.22
JPDA	EnterExitCrossingPaths2	0.963270572	96.32	0.897622605	89.76	0.801649288	80.16	0.87597552	87.59
JPDA	OneLeaveShopReenter1	0.755665869	75.56	0.945655412	94.56	0.947976888	94.79	0.910088436	91.00
JPDA	OneStopEnter1	0.994413722	99.44	0.635405485	63.54	0.832660022	83.26	0.978072467	97.80
JPDA	OneStopEnter2	0.987648763	98.76	0.955463567	95.54	0.996065826	99.60	0.937385701	93.73
JPDA	OneStopNoEnter2	0.996000156	99.60	0.949317177	99.95	0.999514528	99.95	0.781084013	78.10
JPDA	ShopAssistant1	0.944019246	94.40	0.796438533	79.64	0.990531441	99.05	0.97643899	97.64
JPDA	ThreePastShop1	0.989896333	98.98	0.933283141	93.32	0.934251788	93.42	0.676486538	67.64

Tabla 5.4: Coeficiente de Correlación Lineal en el seguimiento empleando el algoritmo JPDA.

Capítulo 6

Conclusiones y Perspectivas

Todo concluye, pero nada perece. Séneca

6.1. Conclusiones

Este trabajo de investigación se realizó para dar seguimiento a personas encontradas en el ángulo de visión compartido por dos cámaras visuales empleando la fusión de datos. Cabe mencionar que en el transcurso de elaboración de este trabajo se pudieron conocer, analizar y programar algunos métodos implícitos y significativos para el entendimiento de los procesos de esta tesis.

El sistema fue diseñado para realizar fusión a nivel características empleando tres algoritmos de asociación de datos: el NNKF, PDA y JPDA, el hecho de ser un trabajo de investigación bastante extenso y poco conocido represento ser un reto difícil de resolver, sin embargo se contaba con los conocimientos básicos para lograr cumplir con los objetivos.

Una de las principales problemáticas fue la detección de los objetos, donde la naturaleza de las imágenes, en distintas ocasiones, no permitió la correcta detección de las personas citadas en el centro de proyección de ambas cámaras, este hecho fue determinante en los procesos consecutivos. Para el algoritmo de la Homografía no se tuvo mucho problema puesto que no fué difícil de implementar ya que se contaba con los puntos de correspondencia de ambas cámaras.

Asi mismo para el análisis y resolución del Filtro Kalman Lineal se necesito realizar en ecuaciones lineales varias pruebas para encontrar los valores adecuados dentro del vector de estados y la matriz de transición y lograr una estimación óptima de los puntos generados.

Para los algoritmos de asociación de datos NNKF y PDA, se realizaron varias pruebas para el mantenimiento de la única trayectoria, en las pruebas realizadas con ecuaciones lineales se obtuvieron mejores resultados, para el caso del JPDA, en el proceso de mantenimiento se le agrego una nueva función que fue el poder generar nuevas

trayectorias a partir de medidas nuevas encontradas en tiempo de ejecución.

Por lo tanto se concluye que los mejores resultados de estimación y asociación de datos se obtuvieron en las funciones lineales obteniendo un 100 %, en la estimación y asociación de datos y con un 90 % obtenido en promedio en las pruebas realizadas en los escenarios, por lo mencionado anteriormente, el algoritmo de segmentación tuvo problemas para detectar los puntos de cada objeto.

Se concluye entonces que se logro el objetivo de asociar trayectorias encontradas en un centro de proyección compartido por dos sensores visuales.

6.2. Perspectivas

Se espera mejorar y robustecer el sistema, para agregar procesos más completos y una utilidad real.

- *Cambiar la plataforma de desarrollo y no dependa de alguna licencia.*
- *Se espera mejorar el algoritmo de detección, para tener un seguimiento de objetos convincente.*
- *Trasladar el sistema a un dispositivo de hardware programable para agilizar el procesamiento de los datos y poder emplearlo en un entorno de tiempo real.*
- *Tener más de dos sensores y de distintos tipos para generar una fusión de información más completa.*
- *No solo realizar el seguimiento de personas, elevar el nivel de fusión de datos a nivel decisión y detectar el comportamiento de los individuos.*
- *Y en el mejor de los casos, detectar los biométricos de cada individuo.*

Muchas son las aplicaciones encontradas en el área de la fusión de datos para investigar y desarrollar proyectos que semi automaticen los procesos de la industria o del hogar. Espero que este trabajo pueda contribuir en un mejor entendimiento y desarrollo de mejores y nuevas tecnologías.

Apéndice A

Modelos de Fusión de Datos

En los años 80 surgieron distintos modelos de fusión con la finalidad de generar sistemas que pudiesen combinar información y obtener mejores estimaciones de los resultados para los sistemas de reconocimiento y seguimiento de objetos, cada modelo desde ese entonces ha definido su arquitectura, sus características y los niveles en los cuales esta dividido, algunos de los modelos se encuentran mencionados en este apéndice por ejemplo: Intelligence Cycle, JDL y el Bucle de Control de Boyd, para 1990 el modelo de Dasarathy y el modelo Waterfall fueron desarrollados y a principios de esta década el modelo Omnibus fué propuesto. Es importante mencionar que estos modelos identifican los procesos, funciones y técnicas aplicables a la fusión de datos como flujos de información desde las fuentes hasta el operador [103], [104], [105].

A continuación se mencionan algunos de los modelos de fusión más importantes.

A.0.1. Modelo Ciclo de Inteligencia o IC, (The Intelligence Cycle):

Se define como un modelo de creación fiable, ya que precisa la inteligencia foránea, es dinámico y sin fin. Este ciclo comienza con preguntas cuyas respuestas generan inevitablemente más preguntas, así que el final de un ciclo, será siempre el comienzo de otro [106], [107], [108], [109]. Este ciclo de inteligencia se compone de cinco fases:

- **Planeación y dirección:** Cuando se tiene la tarea de un trabajo específico, este proceso comienza a planificar lo que se va hacer y cómo. Dirigiéndose a una dirección específica para realizar el trabajo, enumerando lo que se sabe sobre la cuestión y lo que necesita para averiguarlo. Se discuten las maneras de reunir la inteligencia necesaria.
- **Colección:** Este proceso se define como la recolección de datos primarios a partir del cual el término de inteligencia se produce. Hay seis fuentes básicas de inteligencia, o la recaudación de disciplinas:
 - La inteligencia de señales (SIGINT)
 - Imágenes de Inteligencia (IMINT)

- Medición y Firma de Inteligencia (MASINT)
 - Humanos-Fuente de Inteligencia (HUMINT)
 - Fuente Abierta de Inteligencia (OSINT)
 - Inteligencia Geoespacial.
- **Procesamiento:** Implica la conversión de la gran cantidad de información recopilada a una forma utilizable por los analistas a través de la decodificación, traducción de idiomas, y reducción de datos.
 - **Análisis y Producción:** Incluye integrar, evaluar y analizar todos los datos disponibles que son a menudo fragmentarios e incluso contradictorios y en la preparación de productos de inteligencia. Los analistas, examinan la información de fiabilidad, validez y pertinencia. Se evalúa lo que está pasando, por qué está ocurriendo, lo que podría ocurrir y cómo afecta.
 - **Dispersión:** En este último paso, se da el final por escrito a un análisis de políticas, la misma política que inició el ciclo. Después de leer la última instancia de aprendizaje y la respuesta a la pregunta original, las políticas pueden volver con más preguntas. Entonces todo el proceso comienza otra vez.

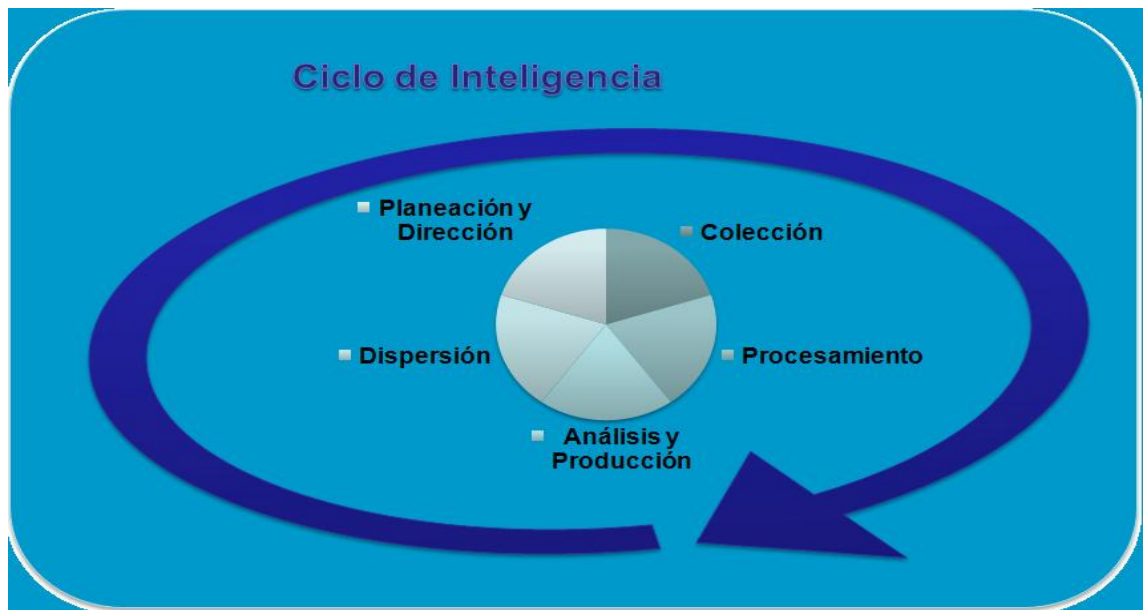


Figura A.1: Modelo de Fusion Ciclo de Inteligencia

A.0.2. Modelo Waterfall (The Waterfall Model):

El modelo Waterfall es un proceso, en el cual el desarrollo es visto como el flujo de una cascada a través de las fases que lo componen. En ocasiones este modelo está definido como un modelo secuencial lineal o como el ciclo de vida del software. El modelo está basado en un artículo publicado por Winston W. Royce, en el que irónicamente el propio Royce lo presenta como un modelo secuencial defectuoso y lo mantiene como un proceso iterativo, la misma crítica generada por el mismo autor ha sido ignorada por mucho tiempo,¹ a pesar de los esfuerzos de Royce, por convertir su modelo en un proceso iterativo, el uso del término Waterfall lo convierte en un modelo de procesos puramente secuenciales en la creación de software que se considera inflexible y reiterativo [110], [111], [112].

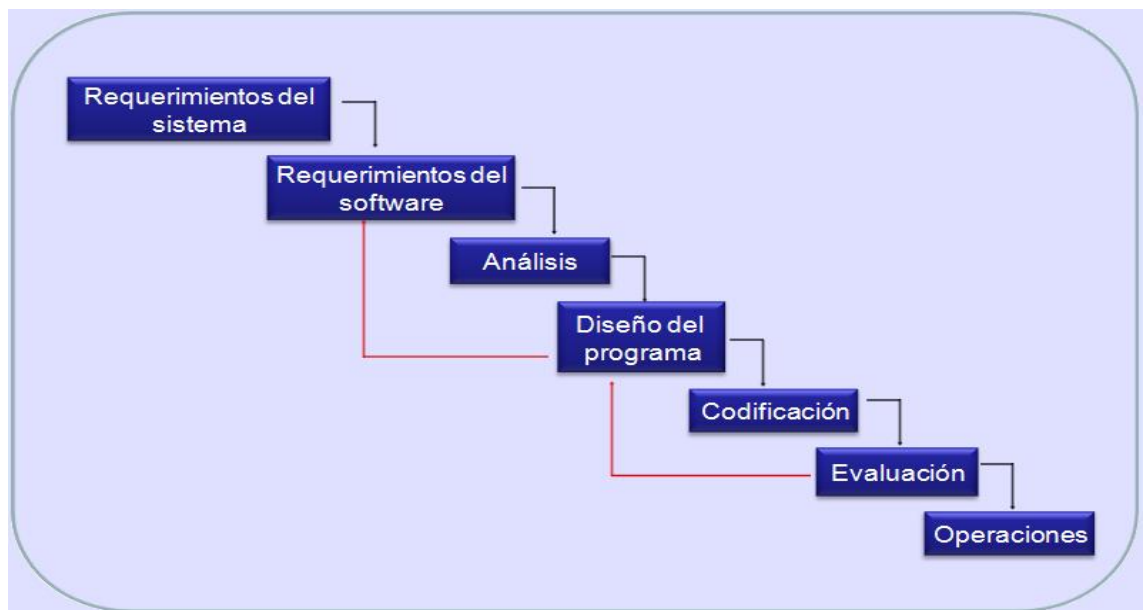


Figura A.2: Modelo de Fusión Waterfall

El modelo está dividido en 6 distintas fases llamadas:

- Fase de Análisis de Requerimientos:
 - a) El problema es ampliamente especificado con los objetivos de los servicios declarados.
 - b) Las restricciones están identificadas.
- Fase de Especificación: Un marco de referencia común para que las descripciones puedan ser coherentes: En esta fase del sistema de especificación se produce des-

¹Un modelo similar al definido por Royce es el llamado Modelo en Espiral

de la definición de detalles sobre a) y, b). En sí, esta fase podría definir claramente la función producto ².

- Fase del Sistema y Diseño de Software: Las especificaciones del sistema se traducen a una representación del software. Se refiere a un ingeniero de software a:
 - *Estructura de Datos*
 - *Arquitectura de datos*
 - *Detalle del algoritmo*
 - *Representación de la Interface*

En esta fase se determinan los requisitos junto con una imagen de la arquitectura del sistema global. Al final de esta etapa, el ingeniero de software debe de ser capaz de identificar la relación entre el hardware, el software y las interfaces asociadas. Cualquier defecto en la especificación podría idealmente no ser aprobado.

- Fase de Implementación y Evaluación: El diseño es traducido a través del dominio del software.
 - *Documentación detallada se puede reducir en la codificación.*
 - *La prueba en esta etapa se centra en asegurarse de que los errores son identificados y que el software cumple con la especificación requerida.*
- Fase de Integración del sistema y pruebas: Todas las unidades del programa son integradas y probadas para asegurar que el sistema completo cumpla con los requisitos de software. Después de esta fase el software se entrega al cliente, para pruebas de aceptación.
- Fase de Mantenimiento: Suele ser la etapa más larga del software ya que aquí el software se actualiza por las siguientes razones.
 - *Satisfacer las cambiantes necesidades de los clientes.*
 - *Adaptado para acomodar los cambios en el entorno externo.*
 - *Corregir errores y omisiones detectados anteriormente en las fases de prueba.*
 - *Aumento en la eficacia de los programas.*

Los bucles de retroalimentación permiten que las correcciones se incorporen al modelo. Por ejemplo, un problema de actualización en la fase de diseño requiere de un “volver” a la fase de especificaciones. Cuando se realizan cambios en cualquier fase, la documentación pertinente debe de actualizarse para reflejar ese cambio.

²En algunos textos el análisis de requerimientos y la fase de especificación son combinadas y representadas como una sola fase

A.0.3. Modelo BCL, (The Boyd Control Loop or OODA Loop):

Este ciclo es aplicado en operaciones de combate, a menudo a nivel estratégico, tanto a nivel militar y en las operaciones comerciales. Fue creado por el estratega militar y USAF el Coronel John Boyd. El ciclo de Boyd esta dividido en 4 etapas. Este ciclo sugiere que la guerra es un modelo constante de evaluación y acción, la parte que pasa a través de cada ciclo más rápido será mejor en constancia que en la realidad. El objetivo según Boyd, es el colapso del sistema adversario en la confusión y el desorden [113], [114], [115].

Los cuatro niveles en los que esta dividido son los siguientes:

- *Observación: Exploración del medio ambiente y obtener información de ella. Anteriormente, las fuentes se limitaron principalmente a lo que podría ser observado ópticamente. La tecnología ha producido toda una gama de sensores, por ejemplo, las bandas del espectro electromagnético (y en el caso de sonar, el espectro sonoro). Tradicionalmente, la observación es un área madura para la creación de fricción en un oponente del ciclo OODA. Atasco, sigilo, camuflaje, señuelos, engaños y otras son todos los esfuerzos para negar una buena observación para el enemigo.*
- *Orientación: Es la parte del ciclo de Boyd donde la información se conjunta durante la observacion, esta es filtrada y organizada. Aquí es donde se decide que es importante y cual es la respuesta para la cuestión. En esta fase se puede tomar ventaja de los engaños y las acciones para tomar cierta ventaja hacia el enemigo, su cultura y su formación. Si se tiene este conocimiento, se puede ver un gran cambio de no mucho esfuerzo.*
- *Decisión: Considera la posibilidad de tener opciones y selecciona posteriormente un curso de acción.*
- *Acción: Lleva a cabo la decisión concebida. Una vez que el resultado de la acción se conserva, para empezar de nuevo. Se tiene que tomar en cuenta que en combate (o compitiendo contra la competencia), se desea ser más rápido que el enemigo, por lo tanto se trata de un buble.*

La fase de Observación es similar al Nivel 0 del modelo JDL y parte de la fase de Coleccion del modelo IC. La fase Orientar incluye las funciones 1, 2 y 3 del modelo JDL y las fases de Coleccionamiento y Recopilación del modelo IC. La fase Decidir es similar al nivel 4 del JDL y las actividades de Dispersion del IC. La fase Actuar³, es el único modelo que se cierra explícitamente en el ciclo, tomando en cuenta el efecto de las desiciones en el mundo real.

³No tiene analogía con el modelo JDL

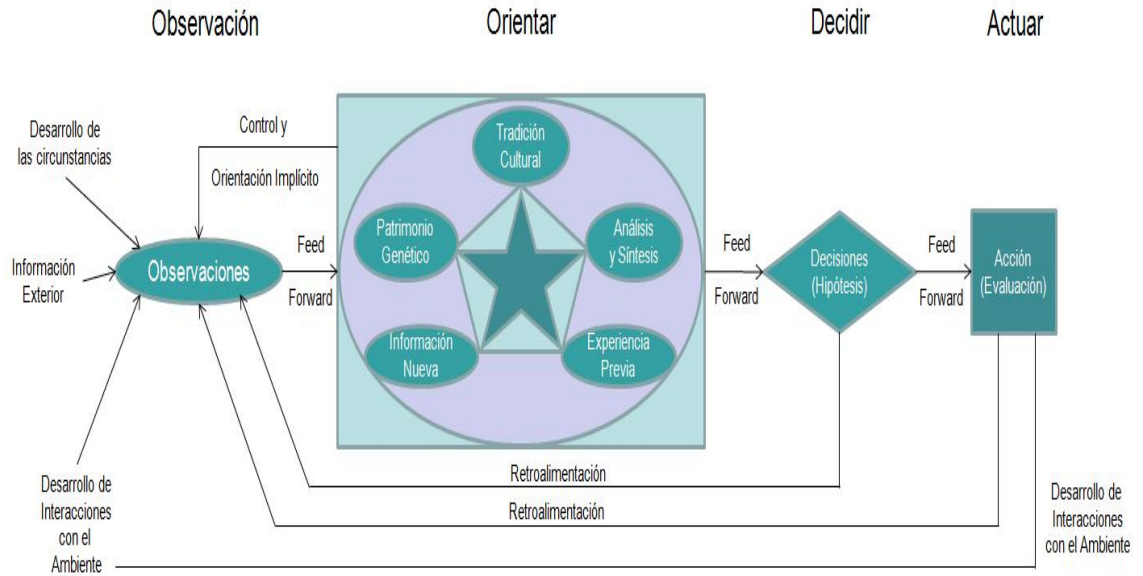


Figura A.3: Modelo de Fusión Boyd

A.0.4. Modelo Dasarthy (The Dasarthy Model):

Se basa más en las funciones que en las tareas y puede ser incorporado en cada una de las funciones de fusión. Muchos investigadores han identificado los tres niveles de abstracción en el proceso de fusión de datos como: la Decisión (símbolos o valores de las creencias), Características (información de nivel intermedio) y Datos (específicamente datos del sensor). Dasarthy señaló que la fusión ocurre dentro de estos niveles como en medio de la transformación de estos.

Dasarthy ha definido una clasificación muy útil de las funciones de la fusión de datos en términos de los tipos de datos e información que se procesan y los tipos que resulten del proceso. Imagen A.5 ilustra el tipos de entradas / salidas consideradas. Los procesos correspondientes a las celdas de la diagonal destaca la región X que descritos por Dasarthy, utilizan las abreviaturas DAI-DAO, la DAI FEO, FEO-FEI, FEI-DEO, y DEI-DEO. Una ventaja notable de esta clasificación es la forma natural en que los tipos de técnica pueden ser asignadas a ella [3].

		Salida					
		Datos	Características	Objetos	Relaciones	Impactos	Respuestas
Entrada	Datos	Detección de la Señal DAI - DAO	Extracción de las Características DAI - FEO	Basada en la Gestalt Caracterización del Objeto DAI - DEO	Basado en la Gestalt Situación de la Evaluación DAI - DEO	Basado en la Gestalt Impacto de la Evaluación DAI - IMO	Respuesta Reflexiva DAI - RSO
	Características	Modelo Basado en la Detección / Extracción de Características FEI - DAO	Refinamiento de las Características FEI - FEO	Caracterización del Objeto FEI - DEO	Basado en la Característica Situación de la Evaluación FEI - RLO	Basado en la Característica Situación de Impacto FEI - IMO	Basado en la Característica Respuesta FEI - RSO
	Objetos	Modelo Basado en la Detección / Estimación DEI - DAO	Modelo Basado en el Fragmento de la Característica DEI - FEO	Refinamiento del Objeto DEI - DEO	Entidad - Relacional Situación de la Evaluación DEI - RLO	Basado en la Entidad Situación de Impacto DEI - IMO	Entidad - Relación Basado en la Respuesta DEI - RSO
	Relaciones	Contexto - Sensible Detección / Estimación RLI - DAO	Contexto - Sensible Extracción de Características RLI - FEO	Contexto - Sensible Refinamiento del Objeto RLI - DEO	Micro / Macro Situación de la Evaluación RLI - RLO	Contexto Sensible Situación de Impacto RLI - IMO	Contexto - Sensible Respuesta RLI - RSO
	Impactos	Costo - Sensible Detección / Estimación IMI - DAO	Costo - Sensible Extracción de la Característica IMI - FEO	Costo - Sensible Objeto de Refinamiento IMI - DEO	Costo - Sensible Situación de la Evaluación IMI - RLO	Costo - Sensible Situación de Impacto IMI - IMO	Costo - Sensible Respuesta IMI - RSO
	Respuestas	Reacción - Sensible Detección / Estimación RSI - DAO	Reacción - Sensible Extracción de la Característica RSI - FEO	Reacción - Sensible Objeto de Refinamiento RSI - DEO	Reacción - Sensible Situación de la Evaluación RSI - RLO	Reacción - Sensible Situación de Impacto RSI - IMO	Reacción - Sensible Respuesta RSI - RSO
		Nivel 0		Nivel 1	Nivel 2	Nivel 3	Nivel 4

Figura A.4: Interpretación de las entradas y salidas del modelo de Fusión de Datos Dasarathy

		Salida			
		Datos	Características	Objetos	
Entrada	Datos	Detección de la Señal DAI - DAO	Extracción de las características DAI - FEO	Caracterización del Objeto Basada en la Gestalt DAI - DEO	
	Características	Detección / Extracción de Características Basadas en el Modelo FEI - DAO	Refinamiento de las Características FEI - FEO	Caracterización del Objeto FEI - DEO	
	Objetos	Detección / Estimación Basadas en el Modelo DEI - DAO	Fragmento de la Característica DEI - FEO	Refinamiento del Objeto DEI - DEO	
		Nivel 0			Nivel 1

Figura A.5: Expansión del modelo de Fusión de Datos Dasarathy desde los niveles 0 al 4.

A.0.5. Modelo Omnibus (The Omnibus Model):

Este modelo es un híbrido de los modelos de fusión: JDL, Ciclo de Boyd, Dasarathy y Waterfall. Este modelo supera algunas de las limitaciones de los modelos anteriores. En la figura siguiente el modelo Omnibus la retroalimentación es explícita y previamente el concepto descuidado del ciclo dentro del ciclo es admitido. La naturaleza cíclica del proceso de fusión de datos se hace explícita mediante la conservación de la estructura general de el ciclo de Boyd. La fidelidad de representación es expresada por el modelo Waterfall, este es fácilmente incorporado a cada una de las cuatro principales tareas del proceso. Los niveles en donde el proceso de fusión toma forma están explícitamente indicados. Por otro lado, el modelo Dasarathy consiste en los tres niveles básicos de fusión de datos: Los datos, función, y la decisión [116].

El modelo de fusión ideal contiene los siguientes puntos:

- *Define el orden de procesamiento.*
- *Hace que el ciclo natural del sistema sea explícito.*
- *Admite la representación de múltiples puntos de vista.*
- *Identifica las ventajas y limitaciones de los diferentes enfoques de fusión.*
- *Facilita la clarificación del desarrollo de las medidas del nivel de tareas de nivel y la eficacia en las medidas a nivel del sistema.*
- *Utiliza una terminología general que es ampliamente accesible;*
- *No asume que las aplicaciones son orientadas a la defensa, (militar).*

El hecho de que esto proporciona una unificación de los modelos señalados anteriormente se desprende claramente del diagrama. Debido a que el modelo de Omnibus se libera de la orientación de los modelos anteriores, no utiliza una terminología específica para aplicaciones militares, ya que incorpora las mejores características de JDL, Dasarathy, OODA y el modelo Waterfall, se sugiere sea la más adecuada (disponible en la actualidad) en el marco para la discusión de las aplicaciones a las estructuras inteligentes. Debido a la inclusión explícita del nivel de control, el modelo de realidad, etc.[117].

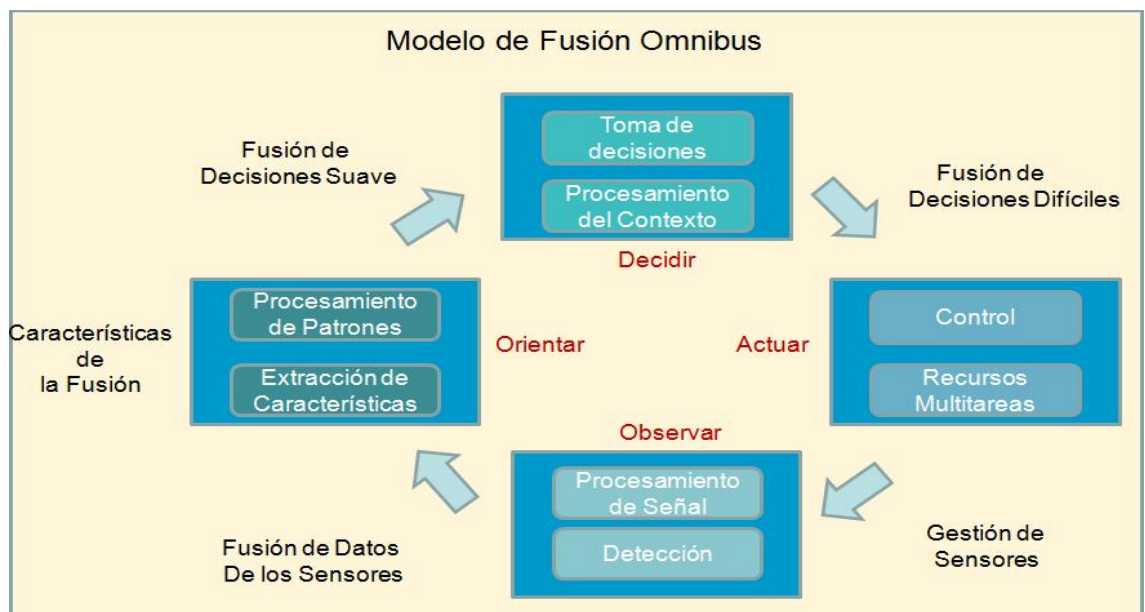


Figura A.6: Expansión del modelo de Fusión de Datos Dasarthy desde los niveles 0 al 4.

Apéndice B

Geometría proyectiva

En esta sección se explican las principales ideas geométricas y notaciones que serán necesarias para la comprensión de algunos términos. La geometría de las transformaciones proyectivas del plano modelan la distorsión geométrica que se introduce sobre un plano cuando se toma una imagen del mismo con una cámara de perspectiva. Algunas propiedades geométricas se conservan, como la colinealidad, por ejemplo, una línea recta se conserva en línea recta; algunas otras propiedades no se conservan, por ejemplo, el paralelismo, ya que las líneas paralelas no se presentan como tales en la imagen ya que estas se ven modificadas. En general la geometría proyectiva modela el proceso de adquisición de la imagen al mismo tiempo que da una representación matemática apropiada a los cálculos.

B.0.6. Plano 2D

En la geometría es común representar un punto en el plano como un par de coordenadas (x, y) en R^2 . De esta manera es fácilmente identificable el plano R^2 . Considerando R^2 como un espacio vectorial siendo el par de coordenadas (x, y) representados como un vector, se logra modelar un punto en el plano como un vector en el espacio 2D.

B.0.7. Puntos y líneas

Un punto en el espacio 2D es representado por las coordenadas (x, y) . La forma de representar una línea en el plano es mediante la ecuación siguiente: $ax + by + c = 0$. Por lo tanto, una línea puede representarse de forma natural por un vector $(a, b, c)^T$. La correspondencia existente entre líneas y vectores $(a, b, c)^T$ no es uno a uno, ya que las líneas $ax + by + c = 0$ y $(ka)x + (kb)y + (kc) = 0$, son la misma para cualquier constante k , distinta de cero. Donde dos vectores relacionados por una escala global son considerados como equivalentes. Una clase de equivalencia de vectores bajo esta relación de equivalencia se conoce como un vector homogéneo. El conjunto de clases de

equivalencia de vectores en $R^3 - (0, 0, 0)^T$ forma el espacio proyectivo P^2 . La notación $-(0, 0, 0)^T$, no corresponde a ninguna línea queda excluido.

B.0.8. Representación homogénea de puntos

Un punto $x = (x, y)^T$ esta sobre la línea $l = (a, b, c)^T$ si y solo si $ax+by+c = 0$. Esta ecuación puede escribirse como un producto interno de vectores, $(x, y, 1)(a, b, c)^T = (x, y, 1)^T \cdot l = 0$; es decir, el punto $x = (x, y)^T$ ha sido representado por un 3-vector añadiendo una tercera coordenada 1. Debemos notar que para cualquier constante k distinta de cero, se seguirá verificando la misma igualdad $(kx, ky, k)l = 0$. Es por lo tanto natural considerar el conjunto de vectores $(kx, ky, k)^T$ para distintos valores de k como la representación del punto $(x, y)^T$ en R^2 . Por tanto al igual que con las líneas y puntos, los puntos se representan por vectores homogéneos. Un vector arbitrario homogéneo representante de un punto es de la forma $x = (x1, x2, x3)^T$, representando un punto $(x1/x3, x2/x3)^T$ en R^2 . Los puntos como vectores homogéneos, son también elementos de P^2 . Del resultado anterior se deduce que un punto x esta sobre una línea l si y solo si se verifica $x^T l = 0$. Para poder especificar un punto es necesario dar dos valores los cuales serán sus coordenadas x e y . De igual manera para especificar una línea habrá que dar dos parámetros (los dos cocientes independientes $a : b : c$) y por lo tanto tiene dos grados de libertad.

B.0.9. Intersección de líneas

Dadas dos líneas $l = (a, b, c)^T$. Se define el vector $x = l \times l'$, donde \times representa el producto vectorial o cruzado. De la identidad del producto escalar triple $l' \cdot (l \times l) = l \cdot (l \times l') = 0$ puede verse que $l'^T x = l'^T$ y $x = 0$. Por tanto si consideramos a x un representante de un punto, dicho punto estará sobre ambas rectas l y l' , por lo tanto en la intersección de ambas.

Este resultado muestra que el punto intersección de dos rectas l y l' esta dado por $x = l \times l'$. De igual manera que el resultado anterior, se puede deducir que el vector que define la recta que pasa por dos puntos x y x' esta dado por $l = x \times x'$.

B.0.10. Puntos ideales y recta del infinito

Uno de los aspectos a tomar en cuenta en la geometría Euclídiana es el poder distinguir entre los puntos del infinito y los puntos finitos. Por ejemplo, en el caso de la intersección de rectas paralelas se conoce que no se tiene solución en el caso Euclídeo y se intersectan en el infinito. En la geometría proyectiva por notación de las coordenadas homogéneas permite abordar el estudio de propiedades de intersección de puntos y rectas sin necesidad de hacer distinción entre los casos finito o infinito.

Como se definió en la subsección anterior la intersección de dos rectas es representada por el producto cruzado de dos rectas, por lo tanto, siempre existe una solución independiente de la situación relativa de las rectas. En el caso de las rectas paralelas,

causa gran expectación en el resultado y no existe gran dificultad ya que en este caso el vector homogéneo representa la solución y obligatoriamente su tercera coordenada será igual a cero, lo que corresponde a un punto fuera del plano R^2 . Por lo tanto los puntos finitos de R^2 están representados por vectores de R^3 con tercera coordenada de $R_3 \neq 0$. Los puntos de R^3 con tercera coordenada igual a cero $x_3 = 0$, se denominan puntos ideales o puntos del infinito. Notese que el conjunto de los puntos ideales, $(x_1, x_2, 0)^T$, está sobre una recta llamada recta del infinito y cuyo vector es $l_\infty = (0, 0, 1)^T$.

B.1. Transformaciones del plano proyectivo

Una proyectividad del plano es una aplicación invertible de P^2 en P^2 , (es decir, de 3 vectores homogéneos) que aplica líneas a líneas. De forma más precisa.

Definición: Una proyectividad es una aplicación invertible h de P^2 en P^2 tal que tres puntos x_1, x_2 y x_3 están alineados si y solo si $h(x_1), h(x_2)$ y $h(x_3)$ lo están.

Las proyectividades forman un conjunto cerrado para la operación inversa y la composición. Una proyectividad también se denomina colineación u homografía. El resultado que permite usar las propiedades algebraicas de una proyectividad es el siguiente:

Teorema. Una aplicación h de P^2 en P^2 es una proyectividad si y solo si existe una matriz de 3 no singular H tal que para cualquier punto en P^2 representado por un vector x es verdad que $h(x) = Hx$.

El teorema asegura que cualquier proyectividad puede representarse como una transformación lineal invertible en coordenadas homogéneas y que inversamente cualquier transformación de este tipo es una proyectividad.

Como consecuencia de este teorema se puede dar la siguiente definición:

Definición. (Transformación proyectiva). Una transformación proyectiva entre planos es una transformación lineal sobre 3-vectores homogéneos x , representada por una (3×3) matriz H , $x' = Hx$.

Es importante resaltar que dado el carácter homogéneo de los vectores, la matriz H puede multiplicarse por una constante sin que la transformación se modifique. Por tanto la matriz H también es de tipo homogéneo y está definida salvo una constante de proporcionalidad. Como consecuencia la matriz H tan solo posee 8 elementos independientes, ya que uno de ellos lo fija la constante de proporcionalidad.

Una transformación proyectiva transforma un plano en otro plano equivalente en el que se conservan todas las propiedades invariantes a las proyectividades.

B.2. Transformación de líneas y cónicas

Como se ha visto la ecuación que define la transformación de los puntos de un plano por una proyectividad está dada por $x' = Hx$, lo que corresponde ahora, es conocer la ecuación que define la transformación de una recta y una cónica. Dado que una proyectividad conserva la colinealidad (ver definición), dada una recta l que contenga

tres puntos x_i también existirá una recta l' en el espacio imagen que contendrá a dichos tres puntos $x'_i = Hx_i$ y cuya relación con l esta dada por $l' = H^{-T}l$

Dado que los tres puntos x_i se verifica que $l^T x_i = 0$, dada una matriz no singular H tenemos que $l^T H^{-1} Hx_i = l^T H^{-1} x'_i = l'^T x'_i = 0$. Luego $l'^T = l^T H^{-1}$.

La ecuación de una cónica en coordenadas homogéneas esta dada por $x^T Cx = 0$. Si tenemos una transformación proyectiva de los puntos del plano $x'_i = Hx_i$, la ecuación de la cónica se transforma de acuerdo a $x'^T C' x' = x^T H^T C' Hx = x^T Cx$. Es decir la transformación por una proyectividad de una cónica plana de matriz C es otra cónica con matriz $C' = H^{-T} C H^{-1}$.

B.3. Jerarquía de transformaciones

Dentro de las transformaciones de proyectividad existe cierta jerarquía o grupo lineal proyectivo, dentro de este grupo de transformaciones existen numerosos subgrupos de gran interés. El grupo de las matrices $n \times n$ invertibles con elementos reales es el grupo lineal general sobre n dimensiones o $GL(n)$. Para obtener el grupo lineal proyectivo es necesario identificar la clase de matrices que son equivalentes salvo una constante, este grupo se nota $PL(n)$ (es un grupo cociente de $GL(n)$). Para este caso $n = 3$.

Los subgrupos importantes de $PL(3)$ incluyen el grupo afín, que es el subgrupo de $PL(3)$ consistente en las matrices para las cuales la última fila es $(0, 0, 1)$, el grupo Euclídeo, que es un subgrupo del grupo afín para el cual la submatriz 2×2 superior-izquierda es ortogonal. También se puede identificar el grupo Euclídeo orientado en el caso en que el determinante de la submatriz es igual a 1.

En las siguientes subsecciones se mostrarán las transformaciones, comenzando por las más especializadas.

B.3.1. Clase I: Isometrías

Las isometrías son transformaciones del plano R^2 que preservan distancia Euclídea. Una isometría se representa por:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} \xi \cos \theta & -\sin \theta & t_x \\ \xi \sin \theta & \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (\text{B.1})$$

Donde $\xi = \pm 1$. Si $\xi = 1$ entonces la isometría preservará la orientación y es una transformación Euclídea. Si $\xi = -1$ la isometría invertirá la orientación (una reflexión). Las transformaciones Euclídeas modelan los movimientos de los cuerpos rígidos. Son las isometrías más importantes y serán objeto de nuestro estudio, pero las isometrías que invierten la orientación también aparecen como posibles ambigüedades en el proceso de recuperación de la estructura geométrica a partir de imágenes.

Una transformación Euclídea plana se puede escribir de forma concisa como sigue:

$$x' = H_E x = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} x \quad (\text{B.2})$$

Donde R es una matriz de rotación 2×2 (es decir ortogonal), tal que $R^T R = R R^T = I$, t es un 2-vector de traslación y 0 es un 2-vector nulo. Casos de especial interés son una rotación pura (cuando $t = 0$) y una traslación pura (cuando $R = I$). A las transformaciones Euclídeas también se les conoce como desplazamientos. Una transformación Euclídea entre planos tiene tres grados de libertad, uno para la rotación y dos para la traslación. Por lo tanto se estiman tres parámetros para definir la transformación. Ya que en la ecuación de la transformación cada correspondencia entre puntos fija dos ecuaciones lineales entre los elementos de la matriz, en este caso con solo dos correspondencias entre puntos es posible calcular los parámetros de la transformación.

Los invariantes de este grupo son bien conocidos: longitudes, ángulos y áreas.

B.3.2. Clase II: Semejanzas

Una semejanza es una isometría compuesta con un escalado isotrópico. En el caso de una representación Euclídea compuesta con una escala, la semejanza tiene la siguiente representación:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} s \cos \theta & -s \sin \theta & t_x \\ s \sin \theta & s \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (\text{B.3})$$

que puede escribirse de forma más compacta como:

$$x' = H_s x = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} x \quad (\text{B.4})$$

Donde s representa el factor de escala. Una semejanza se conoce también como una transformación equiforme ya que conserva la forma. Una transformación de semejanza en el plano tiene 4 grados de libertad (1-escala, 1-giro, 2-traslación), por tanto al igual que la isometría puede ser calculada a partir de la correspondencia entre dos puntos. Los invariantes de esta transformación se pueden construir a partir de los Euclídeos teniendo en cuenta la escala. Por lo tanto, los ángulos son los únicos invariantes Euclídeos que permanecen. También puede observarse que los cocientes entre longitudes y entre áreas son invariantes de esta transformación.

B.3.3. Clase III: Afinidades

Una transformación afín o una afinidad se define como una transformación lineal no singular seguida de una traslación. La representación matricial de la misma es

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (\text{B.5})$$

o en forma compacta:

$$x' = H_A x = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix} x \quad (\text{B.6})$$

Siendo A una matriz 2×2 no singular. Una transformación afín en el plano tiene 6 grados de libertad correspondiendo a los seis elementos de la matriz. Por tanto necesitará de tres correspondencias entre puntos para poder ser calculada. Los dos nuevos grados de libertad aparecen como consecuencia de que en las transformaciones afines se pueden producir deformaciones siguiendo una dirección arbitraria (1 ángulo, 1 parámetro de escala que mide el cociente entre las deformación en la nueva dirección y su ortogonal).

Dado que una transformación afín permite deformaciones no isotrópicas, los invariantes de las semejanzas no lo son de las afinidades. Ahora los invariantes más importantes son: El paralelismo, los cocientes de longitudes de segmentos de líneas paralelas, cocientes de áreas.

Una afinidad preservará o no la orientación del plano, esto en función del signo del determinante de la matriz A . Como $\det(A) = \lambda_1 \lambda_2$ (producto de las deformaciones ejercidas) el signo del determinante dependerá del signo de las deformaciones.

B.3.4. Clase IV: Proyectividades

Las transformaciones proyectivas ya han sido definidas como transformaciones lineales no singulares de coordenadas homogéneas. Evidentemente generalizan las transformaciones afines que son la composición de una transformación lineal general no singular de coordenadas no-homogéneas y de una traslación. La notación compacta de una transformación proyectiva es:

$$x' = H_p x = \begin{bmatrix} A & t \\ v^T & v \end{bmatrix} x \quad (\text{B.7})$$

Donde $v = (v_1, v_2)^T$. La matriz tiene nueve elementos pero solo ocho son independientes. Aunque en muchas ocasiones se fija la escala tomando el valor de $v = 1$, esto no es siempre correcto ya que en algunos casos el verdadero valor de v puede ser 0 y por tanto no ser correcto este escalado. Una proyectividad entre planos puede ser calculada a partir de la correspondencia entre cuatro puntos. Ahora al contrario de las afinidades no es posible distinguir entre proyectividades que preservan o invierten la orientación.

El invariante más importante de las proyectividades entre planos es la razón cruzada de cuatro puntos alineados, que se define como:

$$RC(x_1, x_2, x_3, x_4) = \frac{|\bar{x}_1\bar{x}_2||\bar{x}_3\bar{x}_4|}{|\bar{x}_1\bar{x}_3||\bar{x}_2\bar{x}_4|} \quad (\text{B.8})$$

La principal diferencia entre proyectividades y afinidades es el vector $v = (v_1, v_2)^T$ que define la tercera fila de la matriz de la transformación. En las afinidades este vector es fijo e igual a $(0,0)$ y en cambio en las proyectividades puede ser cualquiera del espacio. Este vector es responsable de los efectos no-lineales de la proyectividad. Para se compara la aplicación de un punto ideal $(x_1, x_2, 0)^T$ bajo una afinidad y una proyectividad. Un punto ideal (un punto del infinito) se proyecta bajo una afinidad en otro punto ideal, mientras que bajo una proyectividad se proyecta en un punto cualquiera del espacio definido por el vector $v = (v_1, v_2)^T$.

B.4. Descomposición de una proyectividad

Una transformación proyectiva puede ser descompuesta en una cadena de transformaciones donde cada matriz de la cadena representa una transformación más alta en la jerarquía que la anterior.

$$H = H_S H_A H_P = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} K & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} I & 0 \\ v^T & v \end{bmatrix} = \begin{bmatrix} A & t \\ v^T & v \end{bmatrix} \quad (\text{B.9})$$

Donde \mathbf{A} es una matriz no singular dada por $A = sRK + tv^T$ y K es una matriz triangular superior normalizada cuyo $\det(K) = 1$. Esta descomposición es válida con tal de que $v \neq 0$ y es única si s se toma positivo. Cada una de las matrices H es de una transformación del tipo indicado por el subíndice. Consideremos el ejemplo ya aparecido de rectificar la deformación de perspectiva de una imagen. En el proceso de rectificación habrá que calcular una transformación proyectiva. Así pues, dicha transformación se puede calcular paso a paso a través de la descomposición anterior. La transformación H_P , (con 2 grados de libertad) mueve el vector $v = (v_1, v_2)^T$ a la recta del infinito, la transformación H_A (2 grados libertad) afecta a las propiedades afines pero no mueve la línea del infinito, la transformación H_S , (4 grados de libertad) es una semejanza general que no afecta ni a las propiedades afines ni a las proyectivas.

Apéndice C

Estimación, Decisión, Seguimiento y otros términos

En este apéndice se muestran algunos términos necesarios y básicos para el entendimiento de los algoritmos de predicción y seguimiento.

Estimación: Es el proceso de inferir el valor cuantitativo de las observaciones que son de interés, (que se pueden dar de manera indirectas e inexactas), puede ser vista como el proceso de selección de un punto desde un espacio continuo.

Decisión: Se puede entender como la selección de un resultado de un conjunto discreto de alternativas, “la mejor elección”, de un espacio discreto. Sin embargo uno puede hablar de estimación en un caso de valores discretos con la posibilidad de no hacer una elección pero obteniendo una probabilidad condicional de varias alternativas. Esta información puede ser usada sin tener problemas de decisión.

Target o Objetivo: Es el blanco o punto de referencia al que se le dara seguimiento, en este caso puede llegar a ser una persona o una cosa.

Medidas: Son observaciones relacionadas al objetivo, pueden estar corrompidas o pueden ser ruido, estan estrechamente relacionadas al estado del objetivo.

Clutter o Interferencia: Es el conjunto de medidas no deseadas que ocasionan confundir el algoritmo, estas se pueden presentar por varios factores, algunos de ellos son por la naturaleza de los sensores o las condiciones atmosféricas generadas en el momento de la obtención de los datos.

Track o Rastreo: Es el estado de la trayectoria estimado desde un conjunto de medidas, los datos que han sido asociados con el mismo objetivo. La razón de los problemas de multi- objetivos es el cumplir con el **proceso de asociación**, para medidas cuyo origen no es seguro.

Tracking o Seguimiento: Se realiza con uno o más sensores, colocados de manera fija o en movimiento, el tracking tiene mayor alcance que la estimación, no solo emplea todas las herramientas desde la estimación, también requiere del uso extensivo de teoría de decisión estadística, especialmente en alguno de los problemas prácticos, esto se nota principalmente en la **asociación de datos**. Consiste principalmente en las siguientes características:

- Componentes kinemáticos: Posición, velocidad, aceleración, velocidad de giro, etc.
- Componentes característicos: Fuerza de la señal radiada, características espectrales, radar de sección transversal, clasificación del objetivo.
- Constante o lenta variación de parámetros; Parámetros aerodinámicos.
- Estimación directa de la posición.
- Componentes característicos: Fuerza de la señal radiada, características espectrales, radar de sección - transversal, clasificación del objetivo.
- Constante o lenta variación de parámetros; Parámetros aerodinámicos.

Filtro: Es la estimación del estado de un sistema dinámico de datos con ruido, esto equivale a un “filtro” de ruido.

Modelo estado-espacio (state-space): Dentro del cual se enmarca la formulación del filtro supone la existencia de una o más variables, o parámetros variables, con los cuales se puede describir completamente el estado en que se encuentra algún sistema en un momento dado. Por eso se llaman variables “de estado”.

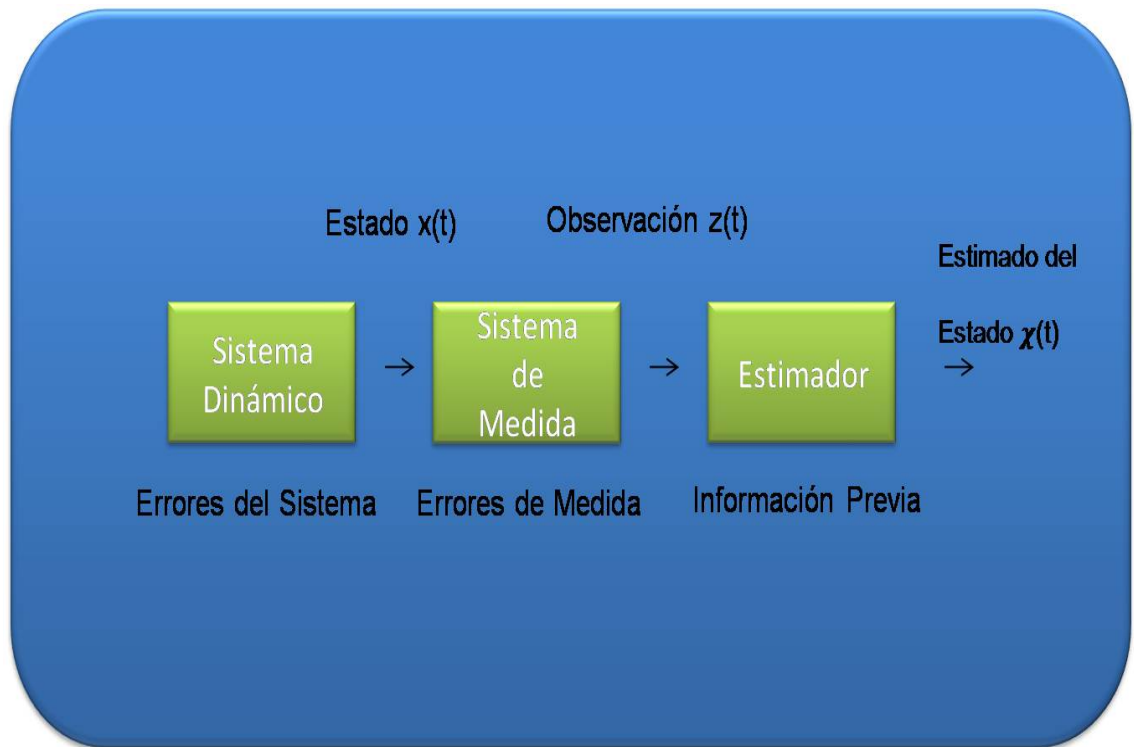


Figura C.1: Formulación matemática del estado de estimación, Estado - Espacio

En la figura mostrada los primeros dos bloques son llamados, “bloques negros”, es decir, no hay acceso de variables dentro de ellos. Las únicas variables válidas para el estimador son las **medidas**, las cuales son afectadas por las fuentes de error en forma de “ruido”.

El estimador emplea el siguiente conocimiento:

- La evolución de la variable, (el sistema dinámico).
- El sensor, (medidas del sistema).
- La caracterización estadística de varios factores aleatorios, (incertidumbres), y la información previa, (apriori).

Estimador óptimo: Es un algoritmo computacional que procesa las observaciones, (medidas), para ceder un estimado de una variable de interés, que minimiza un criterio de error. La ventaja de un estimador óptimo es que se realiza una mejor utilización en los datos y un mejor conocimiento del sistema y las interrupciones, la desventaja es que tiene a ser sensible a modelos de error y puede llegar a ser caro.

Asociación de Datos: Es el proceso que establece que medidas serán usadas en el estado estimador.

La asociación se puede generar de la siguiente manera:

- **Asociación de medida a medida:** En este caso se genera un rastreo o un track.
- **Asociación de medida a rastreo o track:** Se hace un mantenimiento al seguimiento o una actualización.
- **Asociación rastreo a rastreo:** La fusión de los rastreos o seguimientos, para los sistemas de fusión multisensor.

Para realizar la asociación de datos se tienen fundamentalmente dos tipos de algoritmos:

- **Asociación de datos no Bayesiana:** Este enfoque lleva a cabo los procedimientos a nivel de decisión, empleando herramientas estadísticas, (máxima probabilidad o hipótesis evaluada), para después tomar una asociación basada en la decisión, el hecho de que esta no sea correcta se elimina de la secuela.
- **Asociación de datos Probabilística:** Este enfoca la asociación de probabilidades evaluadas y los usa en todo el proceso de la estimación.

Cluster tracking ó Grupo de rastreo: Es el seguimiento de un conjunto de objetos cercanos, preferentemente como un grupo no de manera individual.

Apéndice D

Máquina de Visión Halcon

MVTec es el fabricante de productos como HALCON y ActivisionTools, utilizados en una gran variedad de entornos, tales como industria de semiconductores, medicina, vigilancia y programas de inspección.

MVTec tiene la máquina de visión HALCON, el entorno de programación para el desarrollo de aplicaciones de visión. La flexibilidad de HALCON permite un rápido desarrollo de aplicaciones con un coste reducido. De la misma forma, al ser un entorno fácilmente configurable, permite dar solución tanto a necesidades de visión industrial como de procesamiento de imagen.

HALCON proporciona soluciones de altas prestaciones siendo completamente funcional tanto en avanzadas plataformas Multi-Core como en distintos formatos de hardware de proceso, permitiendo incluso la utilización de HALCON en cámaras inteligentes sin la necesidad de utilizar una PC como unidad de proceso.

Debido a la diversidad en sus librerías que incluyen más de 1400 funciones de procesamiento de imagen, se pueden realizar aplicaciones de visión para el análisis morfológico de objetos, reconocimiento de patrones, códigos de barra/matriz, OCR, aplicaciones de clasificación de color así como una amplia librería de análisis de aplicaciones 3D.

HALCON es totalmente funcional en distintos sistemas operativos entre los que se incluyen Windows (32 y 64 bits), Linux y Solaris. El acceso a las librerías de programación permite, de forma sencilla, generar un código de programa en el lenguaje de programación seleccionado por el usuario (C, C++, C sharp, VisualBasic, .NET y Delphi). Este hecho implica que el usuario de HALCON pueda ser desde un programador con conocimientos básicos hasta un experimentado usuario de librerías de Visión.

HALCON garantiza la independencia del Hardware proporcionando interfaces para distintos estándares de visión (GeniCam, IIDC1394, GigE Vision, etc) permitiendo así la captura de distintos formatos y tipos de cámaras tanto en color como monocromo. De la misma forma el interfaz permite de forma sencilla el desarrollo de aplicaciones multicámara sin ninguna limitación en la resolución de las mismas.

Referencias

- [1] Website. <http://www.tesisenxarxa.net/TDX-0725105-143853/index.html>.
- [2] Wald L. Some terms of reference in data fusion. *IEEE Transactions on Geosciences and Remote Sensing*, Vol. 37(No. 3):1190–1193, 1999.
- [3] James Llinas P. H. D. David L. Hall P. H. D. *Handbook of Multisensor Data Fusion*. 2001.
- [4] Airborne Underwater Geophysical Signals. Data fusion techniques.
- [5] Ahmed Gad and M. Farooq. Data fusion architecture for maritime surveillance. *Department of Electrical and Computer Engineering, Royal Military College of Canada*.
- [6] Duncan Smith and Sameer Singh. Approaches to multisensor data fusion in target tracking: A survey. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, VOL. 18, NO. 12 DECEMBER 2006.
- [7] Website. <http://www.megasociety.org/noesis/167/9.html>.
- [8] Tania Stathaki Ed. *Image Fusion: Algorithms and Applications*. 2008.
- [9] Olivier Faugeras y Quang Tuang Luong. *The Geometry of Multiple Images*. 2001.
- [10] Farhad Samadzadegan. Data integration related to sensors, data and models. *Department of Surveying and Geomatics, Faculty of Engineering*.
- [11] Website. http://people.revoledu.com/kardi/tutorial/KNN/HowTo_KNN.html.
- [12] Website. <http://www.cs.sunysb.edu/~algorithm/files/nearest-neighbor.shtml>.
- [13] Christopher Rasmussen and Gregory D. Hager. Probabilistic data association methods for tracking complex visual objects. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, (No. 6), JUNE 2001.

- [14] Mohammad Bagher Menhaj Aliakbar Gorji Daronkolaei, Vahid Nazari and Saeed Shiry. A joint probability data association filter algorithm for multiple robot tracking problems.
- [15] Guofu Xie Ziqian Zhu Xiaobo Qu, Jingwen Yan and Bengang Chen. A novel image fusion algorithm based on bandelet transform. *Xiamen University, Shantou University, Research Institute of Chinese Radar Electronic Equipment*, March 28 2007.
- [16] R. Joe Stanley. An overview of data fusion methods and applications. *University of Missouri-Rolla*.
- [17] Eds. Siliciano Bruno, Oussama Khatib. *Handbook of Robotics, Sensing and Perception: Sensor Fusion*. 2008.
- [18] L. Wald. Definitions and terms of reference in data fusion. *Groupe Télédétection & Modélisation, Ecole des Mines de Paris*.
- [19] Pramod K. Varshney. Multisensor data fusion. *Department of Electrical Engineering and Computer Science Syracuse University*.
- [20] Sean A. Midwood Murali Tummala and Ian N. Glenn. A multi sensor data fusion algorithm for the uscg's vessel traffic services system. *National Defense Headquarters, Department of Electrical and Computer Engineering*.
- [21] Per Klöör Stan Zyra Erland Jungert, Christian Jönsson. Decision making and data fusion in an interactive adaptive ugs network. *FOA (Swedish Defence Research Establishment)*.
- [22] Website. <http://www.data-fusion.org/article.php?sid=74>.
- [23] Website. http://encarta.msn.com/dictionary_701705479/data_fusion.html.
- [24] Pramod K. Varshney. Multisensor data fusion and applications. *Department of Electrical Engineering and Computer Science Syracuse University*.
- [25] Dr. W. Reynolds Monach. Multi-sensor data fusion system (msdfs). *Daniel H. Wagner Associates, Incorporated*.
- [26] José R. Casar José M. Molina, Jesús García. Analysis of data fusion architectures and techniques in the development of an a-smgcs surveillance prototype. *Universidad Carlos III de Madrid, Universidad Politécnica de Madrid*.
- [27] Ahmed Gad and M. Farooq. Data fusion architecture for maritime surveillance. *Department of Electrical and Computer Engineering, Royal Military College of Canada*.

- [28] Khin Hua Ng Rong Yang Pek Hui Foo, Gee Wah Ng. Application of intent inference for surveillance and conformance monitoring to aid human cognition. *Department of Physics, Advanced Analysis and Fusion Laboratory.*
- [29] Paul Smart Nigel Shadbolt Mark S. Nixon T. Raju Damarla Baofeng Guo, Yi Wang. Approaching semantically-mediated acoustic data fusion. *University of Southampton, U.S. Army Research Laboratory.*
- [30] B. N. Suresh and K. Sivan. Performance evaluation of multi-sensor data-fusion systems in launch vehicles. *Avionics Entity, Vikram Sarabhai Space Centre.*
- [31] Allan Whittaker Buddy H. Jeun. Multi-sensor information fusion technology applied to the development of smart aircraft. *Lockheed Martin Aeronautics Company.*
- [32] Alan N. Steinberg_1 Willem H. Le Roux, Jan J. Nel. Using a data fusion-based activity recognition framework to determine surveillance system requirements. *Council for Scientific and Industrial Research, Independent Consultant_1.*
- [33] Helios Nick McFarlane. Adding value with airport surveillance systems. http://www.eurocontrol.int/airports/public/standard_page/APR1_Projects_ACE.html <http://www.askhelios.com/>.
- [34] Billy Josefsson Constantine Tamvaclis, Nick McFarlane. Use of aircraft derived data for more efficient atm operations. *EUROCONTROL Exp., HELIOS Technology Ltd, LFV (Swedish CAA).*
- [35] H.A.P. Blom E.A. Bloem and F.J. van Schaik. Advanced data fusion for airport surveillance. *Nationaal Lucht en Ruimtevaartlaboratorium, National Aerospace Laboratory NLR.*
- [36] IGNSS Symposium 2006. Stochastic modelling strategies in gps/ins data fusion process. *International Global Navigation Satellite Systems Society.*
- [37] Ali Khaki Sedigh Ali Asadian, Behzad Moshiri and Caro Lucas. Optimized data fusion in an intelligent integrated gps/ins system using genetic algorithm. *World Academy of Science, Engineering and Technology.*
- [38] Dirk Linzmeier. Sensor data fusion: Applications & challenges. *Daimler-Chrysler AG.*
- [39] Anders Grauballe, Gian Paolo Perrucci, and Frank H.P. Fitzek. Introducing contextual information to mobile phones by external and embedded sensors. *Department of Electronic Systems Aalborg University, Denmark.*
- [40] Chia-Hui Chenf Heng-Shuen Chendf Jer-Junn Luhef Hsing-Hau Chenag, Polly Huangab. Sensor data fusion for timely emergency alarm. *National Taiwan University.*

- [41] S. Sitharama Iyengar Krishnendu Chakrabarty Hairong Qi, Xiaoling Wang. Multisensor data fusion in distributed sensor networks using mobile agents. *University of Tennessee, Louisiana State University, Duke University*.
- [42] Website. <http://www.data-fusion.org/article.php?sid=75>.
- [43] Oscar Ricardo Vergara. Data fusion. *Instituto Militar de Engenharia (IME), Departamento de Engenharia Cartográfica (DE/6)*.
- [44] Victor Tom-3 Philip Cheng-1, Thierry Toutin-2. Orthorectification and data fusion of landsat 7 data. *PCI Enterprises-1, Canada Centre for Remote Sensing-2, Atlantic Aerospace Electronics Corp-3*.
- [45] Zeynep Nahide Aydin. The application of multi-sensor remote sensing techniques in archaeology. *The University of Mississippi*.
- [46] L. Cowper R. Gamache-R. Hamm V. Jammu-J. Wagner M. Dhar, S. Bardsley. Diagnostics and data fusion of robotic sensors. *Mechanical Technology Incorporated*.
- [47] Jeremy Thorpe and Robert McEliece. Data fusion algorithms for collaborative robotic exploration. *California Institute of Technology*.
- [48] Jürgen Kittler Calvin R. Maurer Joachim Denzler, Joachim Hornegger. Sensor data and information fusion in computer vision and medicine. *Univ. Jena, Univ. Erlangen, Univ. of Surrey, Stanford Univ*.
- [49] Winston Li Dongliang Huang, Henry Leung. Fusion of dependent and independent biometric information sources. *Department of Electrical & Computer Engineering, University of Calgary*.
- [50] Javier Ortega García Julián Fierrez Aguilar. *Esquemas Adaptados de Fusión para Autenticación Biométrica Multimodal*. Julio de 2006.
- [51] K. K. Benedict T. R. Kuntz R. P. Watson, L. A. Scuderi. Data fusion for environmental assessment: Integrating modis imagery and nexrad weather radar. *University of New Mexico*.
- [52] Lloyd A. Treinish. Coupling of mesoscale weather models to business operations utilizing visual data fusion. *IBM Thomas J. Watson Research Center*.
- [53] The sensor correlation and fusion process. <http://www.au.af.mil/au/awc/awcgate/vistas/>.
- [54] Anthony Kuh T'ulay Adali Udo Trutschell Martin Golz Philippe De Wilde Javier Barria Anthony Constantinides Danilo P. Mandic, Dragan Obradovic and Jonathon Chambers. Data fusion for modern engineering applications: An

overview. *Imperial College London, Siemens AG, University of Hawaii, UMBC, Circadian Technologies, University of Schmalkalden, Cardiff University.*

- [55] Website. <http://www.data-fusion.org/article.php?sid=69>.
- [56] Thomas Kausch Felix Opitz, Werner Henrich. Data fusion development concepts within complex surveillance systems. *Defence and Communications Systems, Air and Naval Defence.*
- [57] Alan Steinberg. Problem-solving approach to data fusion. *Space Dynamics Laboratory, Utah State University.*
- [58] Website. <http://www.data-fusion.org/article.php?sid=70>.
- [59] Galina Rogova Alan Steinberg Ed Waltz James Llinas, Christopher Bowman and Frank White. Revisiting the jdl data fusion model ii. *The 7th International Conference on Information Fusion, June 28 to July 1 2004.*
- [60] Wilfried Elmenreich. Sensor fusion in time-triggered systems. *O. Univ.-Prof. Dr. Hermann Kopetz, Institut für Technische Informatik 182, October 2002.*
- [61] Remco C. de Boer. A generic architecture for fusion-based intrusion detection systems. *Erasmus University Rotterdam, Rotterdam School of Economics.*
- [62] Ronnie Johansson. Information acquisition in data fusion systems. *Royal Institute of Technology, Department of Numerical Analysis and Computer Science.*
- [63] Leonid I. Perlovsky. Cognitive high level information fusion. *Harvard University and Air Force Research Lab., Sensors Directorate.*
- [64] Scott C. McGirr. Resources for the design of data fusion systems. *SPAWAR Systems Center.*
- [65] David L. Hall James Llinas. An introduction to multi-sensor data fusion. *State University of New York at Buffalo, The Pennsylvania State University.*
- [66] Erik. P. Blasch_1 and Susan Plano_2. Jdl level 5 fusion model “user refinement” issues and applications in group tracking. *Wright State University.*
- [67] Atif R. Mirza. An architectural selection framework for data fusion in sensor platforms. *The University of Edinburgh.*
- [68] Jerzy A. Letkowskic_3 Kenneth Baclawskid_4 Mieczyslaw M. Kokara_1, Christopher J. Matheusb_2 and Paul Kogute_5. Association in level 2 fusion. *Northeastern University_1, Versatile Information Systems_2, Western New England College_3, Northeastern University_4, Lockheed Martin_5.*

- [69] Amy L. Murphy Claudionor J. N. Coelho Hervaldo S. Carvalho, Wendi B. Heinzelman. A general data fusion architecture. *University of Rochester, Federal University of Minas Gerais*.
- [70] Robert Willetts Paul Hannah Peter Bryanston-Cross Jaime Esteban, Andrew Starr. A review of data fusion models and architectures: Towards engineering guidelines. *The University of Manchester School of Mechanical, University of Warwick*.
- [71] Gabriel Cristóbal_2 Filip Šroubek_1.2 and Jan Flusser_1. Image fusion based on level set segmentation. *Institute of Information Theory and Automation Academy of Sciences of the Czech Republic_1, Instituto de Óptica_2, CSIC*.
- [72] Website. http://www.ece.lehigh.edu/SPCRL/IF/image_fusion.htm#Image_fusion.
- [73] Guofu Xie_3 Ziqian Zhu _4 Xiaobo Qu _1, Jingwen Yan_2 and Bengang Chen_4. A novel image fusion algorithm based on bandelet transform. *Department of Communication Engineering_1, Department of Electronic Information Engineering_2, Department of Software Engineering_3, Research Institute of Chinese Radar Electronic Equipment_4*.
- [74] Gabriel Cristóbal Salvador Gabarda. On the use of a joint spatial-frequency representation for the fusion of multi-focus images. *Instituto de Óptica, "Daza de Valdés" (CSIC)*.
- [75] Asha Das and K.Revathy. A comparative analysis of image fusion techniques for remote sensed images. *Department of Computer Science, University of Kerala*.
- [76] Anne Solberg. Data fusion for image analysis. *Universitetet I Oslo*.
- [77] Farhad Samadzadegan. Data integration related to sensors, data and models. *Department of Surveying and Geomatics, University of Tehran*.
- [78] Thomas Fechner Oliver Rockinger. Pixel-level image fusion: The case of image sequences. *Autonomous Systems Lab, Intelligent Systems Group*.
- [79] Website. <http://www.sic.rma.ac.be/Research/Fusion/Intro/content.html>.
- [80] Jinwei Zheng Zeeshan Nakhi. Comparison of different image segmentation techniques using active contours. *Autonomous Systems Lab, Intelligent Systems Group*.
- [81] Slav Petrov. Image segmentation with maximum cuts. *University of California at Berkeley*.

- [82] Luis Enrique Sucar. Visión computacional. *Departamento de Computación ITESM Cuernavaca*.
- [83] Luis Gerardo de la Fraga. Segmentación. *CINVESTAV-IPN Sección de Computación. Departamento de Ingeniería Eléctrica*.
- [84] Shobha G. Srinivasan. Segmentation techniques for atrd. *International Journal of Computers*. Issue 2, Volume 2, 2008.
- [85] Ernesto Coto. Métodos de segmentación de imágenes médicas. *Universidad Central de Venezuela, Facultad de Ciencias, Escuela de Computación*. Marzo, 2003.
- [86] David Vernon. *Machine Vision, Automated Visual Inspection and Robot Vision*. 1991.
- [87] Olivier Faugeras and Théo Papadopoulos Quang Tuan Luong. *The Geometry of Multiple Images*. 2001.
- [88] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. 2003.
- [89] Sergio Peralta. *Transformaciones Proyectivas*. 2003.
- [90] Xialo-Rong Li Yaakov Bar-Shalom. *Multitarget-Multisensor Tracking: Principles and Techniques*. 1995.
- [91] Filtro de kalman - busy. <http://iaci.unq.edu.ar/>.
- [92] Marcelino Lázaro. El filtro kalman. *Departamento de Teoría de la Señal y Comunicaciones, Universidad Carlos III de Madrid*. <http://www.tsc.uc3m.es/~mlazaro/>.
- [93] Álvaro Solera Ramírez. *El Filtro de Kalman*. Julio, 2003.
- [94] Álvaro Montenegro García. Introducción al filtro kalman. *Pontificia Universidad Javeriana, Facultad de Ciencias Económicas y Administrativas*. Bogotá, Julio 2005.
- [95] Website. <http://www.cs.unc.edu/~welch/kalman/>.
- [96] Greg Welch and Gary Bishop. An introduction to the kalman filter. *Department of Computer Science, University of North Carolina at Chapel Hill*.
- [97] Maritza de Franco. Una invitación al estudio de las ecuaciones diferenciales ordinarias. <http://www.une.edu.ve/isis/>.

- [98] Alicia Vila Ángel A. Juan, Máximo Sedano. La distribución normal. *Universitat Oberta de Catalunya*.
- [99] Celso Gorrín. Resumen de probabilidad. *Universidad Simón Bolívar*.
- [100] School of Informatics The University of Edinburgh. Information on the yiq color space. <http://hydra.nac.uci.edu/wiedeman/cspace/me/infoyiq.html>.
- [101] Jesús M. de la Cruz García Gonzalo Pajares Martinsanz. *Visión por Computador: Imágenes Digitales y Aplicaciones*. Junio, 2001.
- [102] School of Informatics The University of Edinburgh. Caviar project. <http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/>.
- [103] Jitendra R. Raol. *Multi-sensor data fusion with MATLAB*. Estados Unidos, 2009.
- [104] A Review of Data Fusion Models and Architectures: Towards Engineering Guidelines. Jaime esteban, andrew starr, robert willetts, paul hannah, peter bryanston-cross. *The University of Manchester, University of Warwick*.
- [105] Andres Navarro. Multisensor data fusion applied to augmented reality. *Delft University of Technology*.
- [106] Tech Writer. net. The intelligence cycle. <http://www.tech-writer.net/articles.html>.
- [107] Website. <https://www.cia.gov/kids-page/6-12th-grade/who-we-are-what-we-do/the-intelligence-cycle.html>.
- [108] Website. <http://www.intelligencesearch.com/ia084.html>.
- [109] Alex Burns. Business intelligence: History and the intelligence cycle. Australian Foresight Institute.
- [110] Waterfall model. <http://ocw.mit.edu/NR/rdonlyres/Aeronautics-and-Astronautics/16-355JFall-2005/>.
- [111] Website. http://scitec.uwichill.edu.bb/cmp/online/cs221/waterfall_model.htm.
- [112] Website. <http://www.buzzle.com/editorials/1-5-2005-63768.asp>.
- [113] Frans Osinga. Introducing core ideas and themes of boyd's 'theory of intellectual evolution and growth. <http://www.au.af.mil/au/awc/awcgate/awcgate.htm>.
- [114] Major RA Martinez. Command and control loop relationships: A brief study of marine corps decision-making process and boyd's loop. *EWS Contemporary Issues Paper*, 2006.

- [115] Berndt Brehmer. The dynamic ooda loop: Amalgamating boyd's ooda loop and the cybernetic approach to command and control. *Department of War Studies Swedish National Defence College*.
- [116] Robert Willetts Paul Hannah Peter Bryanston-Cross Jaime Esteban, Andrew Starr. A review of data fusion models and architectures: Towards engineering guidelines. *The University of Manchester, University of Warwick*.
- [117] W.A.; Haywood J. Worden, K.; Bullough. Smart technologies.