



INAOE

Seguimiento de Objetos utilizando Visión Foveal basada en un Nuevo Método de Foveación

por

José Martínez Carranza

Lic., BUAP

Tesis sometida como requisito parcial para
obtener el grado de

**MAESTRO EN CIENCIAS EN LA
ESPECIALIDAD DE CIENCIAS
COMPUTACIONALES**

en el

**Instituto Nacional de Astrofísica, Óptica y
Electrónica**

Enero 2007

Tonantzintla, Puebla

Supervisada por:

Dr. Leopoldo Altamirano Robles

Investigador Titular del INAOE

©INAOE 2007

El autor otorga al INAOE el permiso de reproducir y distribuir copias
en su totalidad o en partes de esta tesis



Resumen

Diversos trabajos han propuesto diferentes métodos para adaptar la visión foveal, sistema inspirado en los sistemas de visión biológicos, a la visión artificial. Los más representativos son la Transformada Log Polar y las Geometrías Cartesianas Exponenciales. No obstante, existen limitaciones inherentes de su misma definición que impiden ampliar su aplicación a una de las tareas más representativas en el área: el seguimiento de objetos.

En este trabajo se presenta un estudio de la visión foveal y su incorporación a la tarea de seguimiento artificial de objetos. A su vez, se presenta una nueva propuesta que permite pasar del espacio invariante al variante bajo los conceptos de la visión foveal. Las imágenes obtenidas con el método de foveación propuesto fueron utilizadas en el seguimiento de objetos sobre escenarios más complejos de los que se han atacado con las propuestas anteriores y se obtuvo un desempeño equiparable al que se puede obtener cuando se realiza el seguimiento en imágenes cartesianas.

Los resultados obtenidos muestran que es posible continuar bajo esta línea de investigación que promueve basar los sistemas de visión artificial en las topologías y mecanismos sugeridos por los sistemas naturales.

Abstract

Several works in machine vision propose different methods to adapt and use a vision system inspired in mammals vision system called foveal vision. Log Polar Transform and Exponential Cartesian Geometries can be found among the most representatives. However, there are important limits in their definition making hard their use in the object tracking task.

This work presents a study concerning with the foveal vision and its application in the object tracking task. Also, a new foveal transform proposal is presented. The images generated with the new foveation method were used in a single object tracking application working with more complex scenes than those treated with the previous proposals. The obtained performance in the tracking is similar to the obtained performance in the tracking task when cartesian images are used.

Results show that it is possible to continue working under this topic concerning with foveal vision and with the key idea of inspiring artificial vision systems on natural topologies and mechanisms.

Agradecimientos

Agradezco al Consejo Nacional de Ciencia y Tecnología CONACYT por la beca que me fue otorgada bajo el número de becario **189903**.

También Agradezco al Instituto Nacional de Astrofísica, Óptica y Electrónica, institución donde realicé mis estudios de maestría con todos los recursos y apoyos necesarios para concluirlos de manera satisfactoria.

Dedicatorias

Es una dedicatoria y agradecimiento para las siguientes personas que han sido fundamentales en mi vida...

Dr. Rodolfo Reyes Sánchez, Dr. Manuel I. Martin Ortiz:

No sólo me enseñaron a buscar ser genuino, a buscar ser de los mejores, a buscar la mejor preparación... No solo me formaron y educaron como soy ahora... También me enseñaron ética, respeto, compromiso y humanidad!... Me enseñaron a respetar y reconocer siempre el trabajo de los demás, a nunca colgarme títulos ni logros que no me correspondieran, me enseñaron que mi trabajo y dedicación es lo que debe hablar de mi y no otros factores, me enseñaron a no abusar de la gente, a no ser del montón, a no ser parte de una mala mayoría, a tener valor y carácter para dar lo mejor de mí...

Con su ejemplo, con la convivencia a su lado, aprendí que tiene un precio el vivir una vida tranquila disfrutando con las cosas que le apasionan a uno... Sin embargo también aprendí que se pueden lograr muchas cosas convirtiéndose uno mismo en ejemplo y hacer ver que no es tan imposible... Espero un día puedan sentirse muy orgullosos de mí y saber que sus enseñanzas, su tiempo invertido, su amistad, no fue en vano... Muchas gracias por todo lo que me han dado...

Dr. Leopoldo Altamirano Robles:

Recuerdo el primer día que lo conocí cuando entré a su oficina y le pedí trabajar con usted, me dio esa oportunidad y confió en mí... Espero haber correspondido de la mejor manera con mi trabajo, con mi dedicación y mi esfuerzo... agradezco tanto sus palabras, su guía, su amistad; lo considero un gran ser humano... Agradezco la oportunidad para trabajar a su lado y la oportunidad de aportar con mis ideas... Me mostró una parte muy importante de la vida que ansiaba conocer y con la que descubrí que es lo que quiero para mi vida... Le agradezco profundamente... Espero un que un día también pueda sentirse orgulloso de mí...

Introducción

Las capacidades de cómputo se han incrementado en los últimos años permitiendo el desarrollo de gran variedad de software. Las capacidades de memoria, velocidades de acceso, velocidad de procesamiento y capacidades de almacenamiento de datos, son las principales propiedades en una computadora que permite el desarrollo de software cada vez más avanzado y con mayores funcionalidades.

La visión por computadora ha sido una de las áreas beneficiadas con los avances tecnológicos pues, en el pasado, la captura y almacenamiento de una imagen digital quedaba restringido a las propiedades descritas anteriormente.

Hoy en día, se cuenta con dispositivos de captura, almacenamiento y procesamiento digitales que permiten llevar a cabo diversas tareas que anteriormente sólo podían ser implementados a nivel hardware. El desarrollo del software para el área de visión, ha representado un factor importante para la reducción de costos de equipo utilizado en visión como cámaras y tarjetas capturadoras. Esto ha significado la apertura de un mercado demandante en soluciones basadas en visión por computadora para resolver diversos problemas como reconocimiento, automatización, control, detección, medición y seguimiento de objetos entre otros.

El seguimiento de objetos es una de las aplicaciones de mayor interés en virtud del mercado que oferta varios problemas a resolver por este medio y los retos que se presenta. Ya sea científico o de desarrollo tecnológico, el seguimiento de objetos con sistemas de visión artificial es un problema iniciado hace más de veinte años y que no ha dejado de representar limitantes.

El primer desafío a vencer es garantizar el seguimiento efectivo de un objeto en el mismo instante en el que el objeto se mueve. Esto implica una fuerte restricción que se resume en el *tiempo de respuesta* que debe tener el sistema para ser capaz de

seguir al objeto de interés y que es conocido como respuesta en tiempo real. El tiempo real significa que el sistema debe responder de acuerdo a un tiempo esperado y sin excederlo de manera que la tarea de seguimiento se pueda llevar a cabo de manera eficaz.

La restricción del tiempo real en los sistemas de seguimiento implican que se efectúe el procesamiento de un número determinado de imágenes por segundo con el fin de obtener la mayor información posible del objeto estudiado.

El tiempo que tarda en procesar una imagen depende de la cantidad de información a procesar de la imagen, entre más elaboradas sean las técnicas y algoritmos utilizados y mayor sea el número de datos que se tengan que analizar en la imagen (píxeles), el tiempo de respuesta se irá incrementando.

Sin importar si las computadoras continúan incrementando sus capacidades de procesamiento, el problema de la respuesta en tiempo real también se ve atado al problema de los recursos disponibles. Se podría pensar en utilizar una super computadora para realizar el seguimiento, pero existirá alguna aplicación en donde se desee llevar a cabo esta tarea con la cantidad mínima de recursos o que tenga que limitarse en los mismos.

De acuerdo al tipo de problemática a resolver y los recursos con los que se cuentan, se tiene que decidir qué algoritmos son adecuados o al menos posibles de utilizar para conseguir un desempeño mínimo y bajo un tiempo de respuesta determinado.

Precisamente, uno de los mayores retos en el seguimiento de objetos y en general, en diversas tareas de visión, consiste en reducir la cantidad de datos a procesar - de ser posible - lo que se reflejará en la disminución del tiempo de procesamiento.

Una disyuntiva emerge ya que se debe establecer un modo de reducir la información sin perder la calidad de la información obtenida por el sensor (la cámara). Por ejemplo, se podría trabajar con imágenes reducidas en resolución a menos de la mitad, pero habrán aplicaciones que exijan un mayor detalle de información el cuál se puede perder gravemente de acuerdo al escalamiento.

La reducción o cualquier estrategia que se relacione con la reducción de datos implica un sub-muestreo de los datos originales. Convencionalmente, este sub-muestreo es invariante y se trata de realizar en proporciones similares para obtener una imagen similar a la original aunque con alguna pérdida de información. El objetivo es mantener la relación espacial original de los píxeles para que los algoritmos se desempeñen

lo mejor posible.

El sub-muestreo invariante como su nombre lo indica, propone realizar un sub-muestreo que varíe o dependa de la zona espacial en donde se muestrea. La idea surgió a partir de las investigaciones realizadas sobre un sistema de visión natural estudiado en la década de los 70's. En estos trabajos se describió la topología y estructura de la retina en los primates. La manera en como se agrupan los fotoreceptores y la cantidad en que se hallan en determinadas zonas, inspiraron el modelo de sub-muestreo invariante que hoy se conoce como sub-muestreo *foveal*, en razón del sistema natural mencionado y que es conocido además como sistema de visión foveal.

En pocas palabras la visión foveal es un sistema cuyas características principales surgen de la estructura que preservan los fotoreceptores en la retina. Al centro, una gran cantidad de fotoreceptores se agrupan formando una región llamada fovea o región foveal y con una capacidad alta de detección. Mientras, alrededor de la región foveal y dirigiéndose hacia las orilla, la presencia de fotoreceptores comienzan a disminuir en una proporción decreciente con respecto del centro: entre más se aleje del centro, menor es el número de fotoreceptores.

Si un pixel es visto como el resultado de la información muestreada por un fotoreceptor, entonces la visión foveal sugiere un sub-muestreo con alta definición al centro mientras que decrece hacia la periferia. Este modo de selección de la información en las imágenes digitales resultó muy apropiado para resolver el problema de reducir datos sin sacrificar resolución o detalle información.

El problema del seguimiento de objeto fue atacado con la propuesta de sub-muestreo foveal bajo la hipótesis de que el problema de seguimiento de un objeto en la escena implica analizar a detalle una región determinada en tanto que se requiere contar también con información con menor relevancia del resto del escenario.

La Transformada Log Polar se derivó de un modelo matemático originado en las investigaciones del sistema de visión foveal y que modela mediante ecuaciones simple la distribución espacial de los fotoreceptores en la visión foveal. Esta fue utilizada posteriormente como el primer método de foveación sobre una imagen digital. El resultado fueron imágenes reducidas en porcentajes mucho menores a un tercio o hasta un cuarto de la información original con información de alta definición en un área de interés y con un sub-muestreo jerárquico del resto del escenario.

Motivación y problemática

La Transformada Log Polar tiene algunas limitantes y el tipo de sub-muestreo da como resultado imágenes foveales que pierden parcialmente la relación espacial de la información, lo que perjudica a la aplicación de los algoritmos para seguimiento. Esto motivo a diseño de algunas propuestas de foveación con diferentes ecuaciones y métodos para el sub-muestreo. El objetivo es experimentar con la forma de realizar el sub-muestreo, siempre basándose en los principios de la visión foveal y para resolver algunos problemas que se generan con un método u otro.

Pero tal pareciera que el auge que cobró esta área fue sólo temporal pues los trabajos más fuertes, relacionados con la visión artificial y el uso de la visión foveal en las tareas como el seguimiento de objetos, se dio en la década de los 90's y desde entonces sólo algunos artículos en conferencias y revistas han tocado el tema, pocos, en comparación con otros temas como detección de movimiento, estimación o medidas de similitud, entre algunos y que se relacionan también con el seguimiento de objetos en tiempo real.

La causa principal de este desinterés se originó por la tecnología misma. La gran mayoría de los sensores trabajan con espacios invariantes de sub-muestreo. Aplicar un método de foveación para cambiar el sub-muestreo variante a uno invariante suponía invertir un tiempo valioso que podría afectar el desempeño de la detección y el seguimiento para las aplicaciones de tiempo real.

La implementación en hardware de estos métodos emergió como una alternativa, pero la transición de software a hardware no era directa puesto que muchos de los métodos utilizaban operaciones y datos no enteros cuya ejecución en hardware es costosa y elaborada.

En suma, los métodos de foveación propuestos hasta la fecha tienen el inconveniente de ser difíciles de implementar en hardware y aunque con otros sea posible, presentan algunos problemas en la relación espacial de los datos al momento del sub-muestreo. La principal problemática en ambos casos es que, para el caso específico del seguimiento de objetos, se restringe el uso de los algoritmos convencionales para el seguimiento. Esto significa que sólo ciertos algoritmos pueden ser aplicados para trabajar con el espacio variante y por lo tanto, las aplicaciones que se han realizado en

cuanto al seguimiento de objetos utilizando la visión foveal, han sido muy específicas y limitadas.

Objetivos

El interés principal de este trabajo es realizar una revisión y estudio de las diferentes propuestas de visión foveal que existen, estudiar sus ventajas, desventajas e investigar de que manera se pueden utilizar o combinar estas estrategias para realizar el seguimiento de objetos con mayores alcances de lo que se ha logrado hasta el momento.

De manera concreta, el objetivo principal de este trabajo de tesis es: *Proponer un método que permita efectuar el seguimiento de objetos utilizando los conceptos de visión foveal.*

Para esto, se tienen los siguientes objetivos particulares:

1. Realizar un estudio de los diferentes métodos de foveación propuestos, analizar ventajas, desventajas y limitaciones.
2. Proponer un nuevo método de foveación que permita superar las limitaciones de los métodos anteriores.
3. Implementar un algoritmo de seguimiento que utilice las imágenes foveales generadas con la propuesta.

Limitaciones

La visión foveal se basa en el proceso de búsqueda y seguimiento de un objeto de interés enfocado en una región de atención. Sobre este objeto de interés se colocará la región foveal en donde el sub-muestreo sea de alta resolución mientras que el resto del escenario es muestreado de manera jerárquica.

Este trabajo se ha restringido para trabajar con secuencias de imágenes obtenidas de escenarios exteriores, los cuales no han sido utilizados en los trabajos previos. Del mismo modo, se restringe al uso de secuencias obtenidas con cámara fija, fondo fijo o cambios graduales en el escenario.

Organización de la tesis

La presente tesis se ha organizado de la siguiente manera:

- En el capítulo 1 se realiza una descripción de los conceptos naturales de visión foveal que sirvieron como inspiración para los métodos de visión foveal artificiales.
- En el capítulo 2 se expone una investigación de los diferentes trabajos que se han propuesto para implementar los conceptos foveales en la visión artificial y orientado a su uso en la tarea de seguimiento de objetos.
- En el capítulo 3 se describen los métodos utilizados para llevar a cabo el seguimiento de objetos y la descripción de cómo se aplican en las imágenes foveales propuestas.
- En el capítulo 4 se presenta un método de foveación que se propone basado en el estudio de los métodos existentes.
- En el capítulo 5 se presentan los experimentos y evaluaciones correspondientes que sustentan este trabajo.
- En el capítulo 6 finalmente se presentan las conclusiones a las que se ha llegado, las limitaciones y trabajo a futuro.

Adicionalmente este trabajo cuenta con dos apéndices. El primero presenta los artículos publicados en memorias de conferencias internacionales que se generaron de este trabajo. El segundo describe la implementación en hardware del método de foveación propuesto, las estrategias utilizadas así como las herramientas de trabajo.

Índice General

Introducción	I
Lista de Figuras	XI
Lista de Tablas	XV
1. La visión foveal y el sistema de visión humano	1
1.1. El campo de visión con resolución uniforme	1
1.2. El sistema de visión de los primates	3
1.3. El sistema de visión humano	4
1.3.1. Zona o región foveal	6
1.3.2. Zona o región periférica	7
1.4. La fijación y el reconocimiento de formas	7
1.4.1. Fijación y movimientos sacádicos	10
2. Uso de la visión foveal en la visión por computadora (seguimiento de objetos)	13
2.1. El seguimiento de objetos	13
2.1.1. Regiones o ventanas de procesamiento	14
2.2. La visión foveal en la tarea de seguimiento de objetos	15
2.3. Primer modelo matemático: la transformada log polar	17
2.3.1. Propiedades	21
2.3.2. Ventajas	26
2.3.3. Desventajas	29
2.3.4. Aplicaciones	30

2.4.	Geometrías cartesianas exponenciales	33
2.4.1.	Análisis piramidal de imágenes	34
2.4.2.	El polígono foveal	35
2.4.3.	Ventajas	37
2.4.4.	Desventajas	37
2.4.5.	Aplicaciones	38
2.5.	La transformada wavelet foveal	38
2.6.	Vision foveal y periférica mediante el uso de cámaras de diferente resolución	40
3.	Algoritmos para el seguimiento	43
3.1.	Especificaciones del escenario	43
3.2.	Detección de movimiento	45
3.2.1.	Método de diferencias de imágenes	45
3.2.2.	Extracción y modelado del fondo	47
3.3.	Detección del objeto de interés	50
3.3.1.	Métodos basados en características	50
3.3.2.	Métodos basados en área	52
3.4.	Ventanas de búsqueda	55
3.5.	Estimación de la trayectoria	57
3.6.	Algoritmos para seguimiento	61
3.6.1.	Algoritmo de seguimiento básico: detección con medida de similitud	61
3.6.2.	Algoritmo de seguimiento: medida de similitud y detección de movimiento	62
3.6.3.	Algoritmo de seguimiento con estimación de trayectorias	64
3.6.4.	Algoritmo de seguimiento para imágenes foveales	65
3.7.	Discusión	68
4.	Método de foveación propuesto	71
4.1.	Modelo log polar discreto	72
4.2.	Foveación con anillos cuadrados	74
4.2.1.	Foveación con el método propuesto	79

4.2.2.	Foveación inversa con el método propuesto	80
4.3.	Algoritmos de foveación simple y promedio	82
4.3.1.	Foveación simple	83
4.3.2.	Foveación promedio	84
4.3.3.	Foveación inversa y recuperación de la imagen	85
4.4.	Discusión	85
5.	Experimentos y resultados obtenidos	87
5.1.	Configuración para la foveación	87
5.2.	Foveación simple y promedio	91
5.3.	Detección de movimiento	93
5.3.1.	Experimentos	95
5.3.2.	Discusión	101
5.4.	Seguimiento de un objeto en la escena	102
5.4.1.	Seguimiento con fovea fija	103
5.4.2.	Seguimiento con fovea móvil	104
5.4.3.	Experimentos	106
5.4.4.	Discusión	117
6.	Conclusiones	121
6.1.	Conclusiones finales	121
6.2.	Limitaciones	123
6.3.	Perspectivas	124
A.	Artículos publicados	125
B.	Implementaciones en hardware para la foveación	127
B.1.	La transformada log polar	128
B.1.1.	Sensores CCD log polares	128
B.1.2.	Implementación FPGA	130
B.2.	Geometría cartesiana exponencial	132
B.3.	Implementación FPGA para el método de foveación propuesto	132
B.3.1.	Estrategia de implementación	132
B.3.2.	Uso de tablas precalculadas	135

B.3.3. Traducción de operaciones en componentes de hardware . . .	135
B.3.4. Multiplicador con estrategia pipeline	136
B.4. Resultados obtenidos	139
B.5. Discusión	141

Lista de Figuras

1.1. Campo de visión uniformemente muestreado	2
1.2. Campo de visión con resolución uniforme y campo de visión con resolución decreciente	4
1.3. Distribución de los conos y bastones a lo largo de la retina [?]	6
1.4. Relación de la distribución de los conos y bastones a lo largo de la retina [?]	8
1.5. Ejemplo de procesamiento foveal y periférico [?]	9
2.1. Distribución de los fotoreceptores de acuerdo a Schwartz	18
2.2. Mapeo del plano Cartesiano al plano Log Polar	20
2.3. Aplicación de la Transformada Log Polar para mapear una imagen digital en una imagen Log Polar	21
2.4. División de la zona foveal y la zona periférica en la imagen Log Polar	23
2.5. Invarianza de las imágenes Log Polares ante la rotación y el escalamiento	25
2.6. Efectos producidos sobre la imagen Log Polar al rotar y escalar la imagen digital que se muestrea	27
2.7. Construcción de una pirámide de imágenes para un análisis jerárquico	34
2.8. Polígono foveal y la construcción de la pirámide bajo el concepto de las Geometrías Cartesianas Exponenciales	36
2.9. Ejemplo de la construcción de la pirámide multiresolución usando el polígono foveal sobre una imagen digital	37

2.10. Aplicación de la Transformada Wavelet Foveal, a la izquierda imagen original, al centro descomposición piramidal utilizando la Transformada Wavelet con los filtros y a la derecha, reconstrucción de la imagen con el efecto foveal	39
2.11. Sistema estéreo de 4 cámaras, 2 cámaras periféricas y 2 foveales . . .	40
3.1. Diferencia de imágenes para detectar movimiento, en orden: imagen anterior, imagen actual, diferencia de actual con anterior, y diferencia absoluta	46
3.2. Detección de objetos en movimiento utilizando el promedio y la desviación estándar de las observaciones realizadas sobre los pixeles. La imagen de la izquierda corresponde a la imagen original, la del centro a la detección realizada por el modelo y la derecha a la detección que se le muestra al usuario en la interfaz.	49
3.3. Detección de un objeto utilizando un modelo basado en el área	53
3.4. Delimitación del área de trabajo mediante el uso de una ventana de búsqueda denotado con el color amarillo	56
3.5. Diagrama a bloques de un algoritmo de seguimiento básico	62
3.6. Diagrama a bloques del algoritmo de seguimiento en imágenes cartesianas	63
3.7. Diagrama a bloques del algoritmo de seguimiento en imágenes cartesianas con un filtro Kalman incorporado para estimación de trayectoria	64
3.8. Diagrama a bloques del algoritmo de seguimiento incorporando el uso de las imágenes foveales	65
3.9. Ejemplos que muestran cual es el efecto en la imagen foveal de que un objeto (mancha negra) se encuentre localizado en diferentes regiones sobre el escenario	69
4.1. Estructura de la imagen foveal a la izquierda y cómo se lleva a cabo el muestreo en la imagen original	76
4.2. Imagen Original, imagen que muestra el muestreo foveal basado en anillos cuadrados, imagen resultado de la foveación e imagen recuperada a partir de la imagen foveal	82

4.3. Ejemplo del proceso de foveación propuesto, a la izquierda la imagen original a muestrear, en el centro la imagen foveal y en la derecha, la imagen recuperada con la transformación inversa de la foveación . . .	85
5.1. Cálculo de las dimensiones del objeto más pequeño que podría ser muestreado por el método de foveación de anillos cuadrados	88
5.2. Ejemplos de foveación obtenidos de diferentes configuraciones en los parámetros de la tabla 5.1	90
5.3. Comparación de las imágenes foveales generadas mediante la foveación promedio y la foveación simple	92
5.4. Detección de movimiento en la secuencia MeetSplit3rdGuy Cartesiana y foveal	99
5.5. Detección de movimiento en la secuencia WalkbyShop1cor en Cartesiano y foveal	100
5.6. Detección de movimiento en la secuencia Carretera en Cartesiano y foveal	102
5.7. Seguimiento de un objeto sobre la imagen foveal con fovea fija	103
5.8. Comparación del error de posición del seguimiento sobre imágenes foveales obtenidas con la foveación promedio contra el seguimiento en imágenes foveales obtenidas con la foveación simple	105
5.9. Secuencia 1	109
5.10. Resultados obtenidos para el seguimiento en la secuencia 1	109
5.11. Secuencia 2	110
5.12. Resultados obtenidos para el seguimiento en la secuencia 2	111
5.13. Secuencia 3	113
5.14. Resultados obtenidos para el seguimiento en la secuencia 3	113
5.15. Secuencia 4	115
5.16. Resultados obtenidos para el seguimiento en la secuencia 4	115
5.17. Secuencia 5	117
5.18. Resultados obtenidos para el seguimiento en la secuencia 5	118
B.1. Sensor CCD Retinal propuesto por Sandini a la izquierda y a la derecha un detalle de la región foveal	128

B.2. Sensor CCD Retinal que captura color, la imagen de la izquierda muestra esquemáticamente la colocación de sensores para adquirir cada canal de color RGB	129
B.3. Sensor CCD Retinal propuesto por Sandini a la izquierda y a la derecha un detalle de la región foveal	130
B.4. Sensor polar logarítmico desarrollado por la universidad de McGill . .	131
B.5. Diagrama de componentes digitales de la arquitectura FPGA del algoritmo de foveación inversa	136
B.6. Esquema del multiplicador construido de un arreglo de multiplicadores para producir resultados en pipeline	137
B.7. Ejemplos de imágenes foveales obtenidas con la arquitectura propuesta	140
B.8. Arquitectura diseñada con las librerías de Xilinx para MatLab	140

Lista de Tablas

4.1. Indexado de los anillos en la imagen cartesiana a los anillos de la imagen foveal	78
5.1. Parámetros de configuración para el método de foveación	89
5.2. Parámetros de configuración para las imágenes de la figura 5.2	91
5.3. Configuración del algoritmo para la detección de movimiento	95
5.4. Secuencias de prueba para la detección de movimiento	96
5.5. Parámetros de configuración para la foveación en los experimentos realizados	108
B.1. Parámetros de configuración para el método de foveación	139
B.2. Configuración para los experimentos realizados y similitudes obtenidas	140
B.3. Datos obtenidos por el reporte Place and Route	141

La visión foveal y el sistema de visión humano

Como muchas de las teorías y técnicas desarrolladas en las ciencias computacionales, este trabajo inspira sus principales ideas en conceptos biológicos como es el caso del sistema de visión foveal. En este capítulo se explican los conceptos necesarios para comprender qué es la visión foveal y cuales son sus principios. Cabe resaltar que la descripción de los fundamentos biológicos no es profunda pero si lo suficiente para comprender el funcionamiento de este sistema natural y poder intuir su uso como fuente de inspiración para aplicaciones en la computación.

1.1. El campo de visión con resolución uniforme

Uno de los cinco sentidos más valorados entre los humanos es el sentido de la vista. La percepción del mundo y la adquisición de conocimiento a través de la apreciación visual de éste, es una de las tareas que se realizan con mayor frecuencia. De ahí que una de las necesidades de mayor importancia ha sido la captura visual de un escenario. Bosquejos, dibujos, pinturas, fotografías y posteriormente, medios más sofisticados de captura temporal como películas y hasta películas con sonido integrado, son los medios que se han creado para lograr el objetivo de archivar una imagen visual.

Actualmente existen medios muy sofisticados para captura de imágenes visuales como las cámaras digitales que no sólo poseen ya grandes capacidades de almacenamiento, sino también captura de imágenes con un *campo de visión amplio* así como con una *alta resolución*. El campo de visión puede entenderse como la porción del

escenario que queda capturada en la imagen; y la resolución, en términos computacionales, significa la cantidad de píxeles que conforman la imagen digital.

Un aspecto importante en las cámaras digitales convencionales es que la resolución es uniforme sobre el campo de visión. Esto es, el campo de visión se encuentra dividido exactamente en una malla o matriz de píxeles en la cual se almacena de manera proporcional una porción del campo de visión. Por ejemplo, una imagen digital de 512×512 píxeles implica que el campo de visión ha sido dividido en una malla de 512×512 puntos o unidades, cada una de las cuales contiene una muestra del campo de visión real de manera equiespaciada y es transformada a una representación numérica interna que la computadora maneja para poder almacenarla y posteriormente manipularla. La figura 1.1 muestra el ejemplo de un objeto que es digitalizado con una distribución de píxeles en forma de malla.

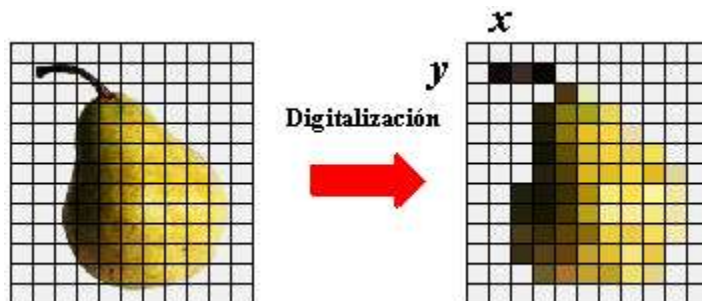


Figura 1.1: Campo de visión uniformemente muestreado

El método de resolución uniforme con la que las imágenes son capturadas en una cámara digital convencional podría pensarse como un modo natural de captura. El campo de visión capturado aparece uniforme a lo largo de la imagen y con esto se hace referencia a que no hay zonas más nítidas que otras. No existen zonas borrosas, ni hay información difusa en la imagen. Esto es, idealmente el muestreo uniforme que realiza el sensor de la cámara digital debería devolver una imagen tal y como la percibe el sentido de la vista.

1.2. El sistema de visión de los primates

Cuando las cámaras digitales surgieron, la idea de copiar el mecanismo de captura visual de la naturaleza emergió. De manera un poco diferente al concepto manejado anteriormente con la cámara fotográfica convencional, la cámara digital se construye de un conjunto de sensores de tipo CCD (Charge-Coupled Device). Cada uno de estos sensores toma el papel similar al de un fotoreceptor, célula del ojo humano encargada de percibir la luz para enviarla posteriormente a la zona cerebral correspondiente, en donde se reúne y procesa la información recibida por los todos los fotoreceptores.

La ventaja de un dispositivo electrónico capaz de almacenar la información de una imagen de manera digital es notoria. Diversas aplicaciones computacionales se han desarrollado; áreas como el procesamiento de imágenes digitales y la visión por computadora han cobrado mucho auge. Cada una de estas áreas establece como hipótesis que, emulando al sistema de visión humano, es posible resolver y automatizar diversas tareas que podrían ser complejas de resolver en poco tiempo para el mismo humano. Para emularlo se han desarrollado teorías y métodos que tratan de copiar o que se inspiran de los procesos naturales de visión. Sin embargo, un detalle curioso es que la distribución de los fotoreceptores sobre la retina no es exactamente una distribución uniforme ni en forma de malla como lo son los sensores CCD de algunas cámaras digitales.

En 1977, en un trabajo publicado por Schwartz [?], se presentó un modelo matemático para describir la distribución de los fotoreceptores que componen la retina de los primates. Este estudio resultó muy importante ya que se sabe que el primate es un pariente muy cercano del humano y muchos de los sistemas biológicos son muy similares entre sí. De este modo, la investigación mostró que el campo de visión que el ojo percibe no es uniforme y de hecho, mostró también que la información que ingresa al cerebro por parte de los fotoreceptores es mucho menos de la que se cree que se aprecia.

El trabajo de Schwartz describió la distribución de las células fotoreceptoras sobre la retina. Esta distribución obedece a una alta concentración de fotoreceptores sobre el centro del campo visual, mientras que esta proporción decrece cuanto más uno se va alejando del centro. Esto implica que en vez de enviar al cerebro una imagen con

un campo de visión uniforme, se envía una imagen con un campo de visión con alta resolución al centro y una resolución difusa a los alrededores. La figura 1.2 muestra un ejemplo de una imagen que ilustra cómo se ve un objeto por una retina cuyos fotoreceptores se encuentran uniformemente distribuidos, caso convencional de las cámaras CCD. En la misma imagen a la derecha, se muestra cómo se ve el mismo objeto por una retina cuyos fotoreceptores se encuentran distribuidos de manera no uniforme, y con una resolución mayor al centro; combinada con una resolución decreciente hacia los bordes. Esta última es la imagen que la retina percibe y envía al cerebro para su procesamiento según el modelo de Schwartz.

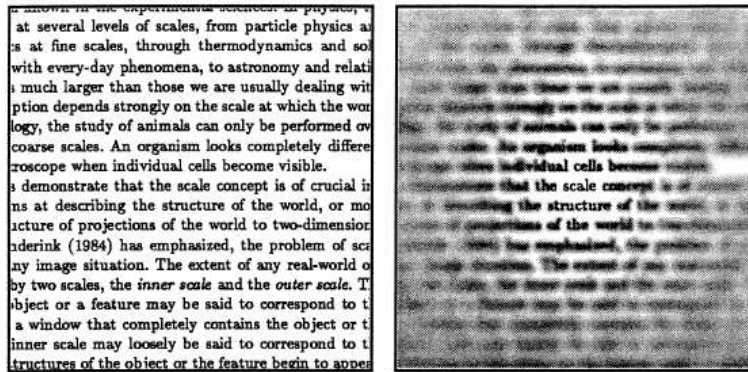


Figura 1.2: Campo de visión con resolución uniforme y campo de visión con resolución decreciente

A este tipo de sistema de visión se le denomina sistema de visión foveal ya que a la zona con mayor concentración de fotoreceptores se le denominó zona foveal y a la zona alrededor del centro o zona foveal con una concentración decreciente de éstos, se le denominó periferia.

1.3. El sistema de visión humano

Si no se tuviera conocimiento de que el ojo no tiene una resolución uniforme, probablemente se tendería a pensar que las imágenes que se producen en el cerebro tienen un campo uniforme como el de cualquier cámara. Sin embargo, en estudios similares a los de Schwartz se ha mostrado que, si el ojo se conformara de un número uniforme de fotoreceptores para producir una imagen con campo de visión

uniforme, entonces el cerebro humano tendría que aumentar tanto su capacidad de procesamiento como de volumen por lo que, el tamaño del cerebro sería diez veces más grande. La pregunta es por qué sin importar el tamaño del cerebro, el ojo no posee una resolución uniforme en los fotorreceptores. Para conocer la respuesta es necesario conocer cómo trabaja el sistema de visión.

El sistema consta de tres etapas básicas:

1. Receptor: Equivalente al ojo y más específicamente a la retina que contiene a las células fotorreceptoras.
2. Vías sensoriales: Nervio óptico que conecta a las células fotorreceptoras con la unidad de procesamiento.
3. Centro de procesamiento: Sección del cerebro encargada de analizar la información transmitida por el receptor.

El ojo tiene la capacidad por un lado de enfocar las imágenes como estímulos luminosos en forma nítida a través de la córnea, el cristalino y los medios transparentes (humor acuoso y humor vítreo) sobre la retina.

La retina es el transductor, es decir, aquella porción del receptor que transforma los estímulos luminosos en una señal capaz de ser leída como información por el cerebro, esto es, estímulo visual transformado en estímulo eléctrico. La retina corresponde tanto en su estructura como en su origen embrionario a una prolongación de la corteza cerebral, por lo que está compuesta de terminales neuronales o células nerviosas dispuestas en una intrincada red similar a los circuitos de un computador [?].

Los estímulos transformados de luz en corriente eléctrica viajan por las membranas celulares de las células nerviosas a través del nervio óptico haciendo cruces y sinapsis (unión) a través de las vías ópticas del cerebro, a través de la red neuronal de las zonas primarias y secundarias del cerebro que analizan la información y dan la sensación de una imagen.

Dentro de estas células existen dos tipos específicos que se han especializado en excitarse con el estímulo luminoso y producir un estímulo eléctrico o nervioso; éstos

son los *conos* y *bastones* y corresponden a los transductores. Estas células se encuentran distribuidas a lo largo de la retina de manera no uniforme y la manera en como se encuentran distribuidas caracteriza y divide a la retina en dos importantes zonas: la zona de la fovea y la zona periférica.

1.3.1. Zona o región foveal

Los *conos* (ver la figura 1.3) se encargan de la percepción de los colores, ya que tienen pigmentos capaces de excitarse con longitud de onda con los colores primarios y están concentrados mayoritariamente en la parte central de la retina o mácula (de unos 500 micrones de diámetro) que concentra miles de estas células por lo que tiene una alta definición y una gran capacidad de representación en el cerebro, además requieren intensidades altas de luz para funcionar. A esta zona con alta concentración de conos y ausencia de bastones se le conoce como *fovea* o *región foveal*.

Esta zona es de vital importancia para un reconocimiento a detalle de objetos y formas gracias a la densa concentración de conos que, al fungir como receptores de información, proveen una mayor resolución y por lo tanto un detalle más fino de la superficie que se está observando.

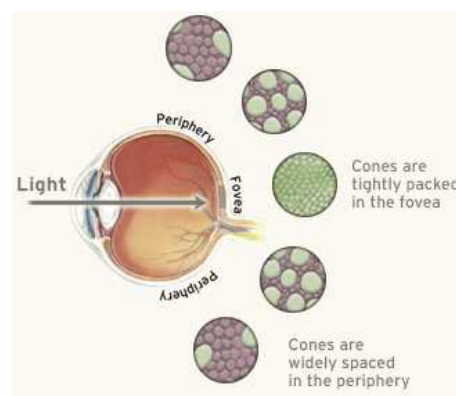


Figura 1.3: Distribución de los conos y bastones a lo largo de la retina [?]

1.3.2. Zona o región periférica

Los *bastones* (ver la figura 1.3) están distribuidos en el resto de la superficie retinal más distanciados entre sí que los conos con menor representación a nivel cerebral que ellos, por lo que no son capaces de resolver detalles y su discriminación es muy baja. No tienen los pigmentos necesarios para distinguir colores; éstos se estimulan con bajas intensidades de luz. El tamaño de los bastones es mayor en comparación de los conos. Su distribución se realiza a lo largo de la retina y como es el caso inverso, no existe presencia de éstos en la zona foveal. La zona en donde se encuentran distribuidos los bastones es conocida como zona o región *periférica*.

A lo largo de la zona periférica es posible encontrar algunos conos, pero conforme se llega a los extremos de la retina, su presencia disminuye al punto de ser casi nula. La figura 1.4 muestra la relación que existe entre los conos y los bastones en cuanto a su distribución en la retina. Como puede observarse, la densidad de conos es mayor en el centro, es decir en la región foveal. Esta densidad se muestra en el pico de la curva más oscura, mientras que la densidad de bastones aumenta a lo largo de la zona periférica mostrado en la curva más clara.

Como se ha expresado anteriormente, las células fotorreceptoras o conos ubicados en la zona foveal o en el centro de la retina, tienen una importancia significativa al ser éstos los receptores que brindan un alto nivel de información y detalle de la escena observada. Sin embargo, es curioso indicar que el ojo humano en promedio posee 6 millones de conos contra 125 millones de bastones que existen en total a lo largo de la retina. Estos datos junto con la distribución expresada en la figura 1.4 muestran la similitud de la distribución de las células fotorreceptoras en comparación con la distribución en los primates propuesta por Schwartz.

1.4. La fijación y el reconocimiento de formas

La distribución de los fotorreceptores como se explicó en la sección anterior no es arbitraria. Si bien el número de bastones es notablemente mayor que el de conos, el área que deben cubrir en la periferia es mucho mayor que la región foveal. Por ese motivo el tamaño de los bastones es mayor y su proporción en número también es mayor.

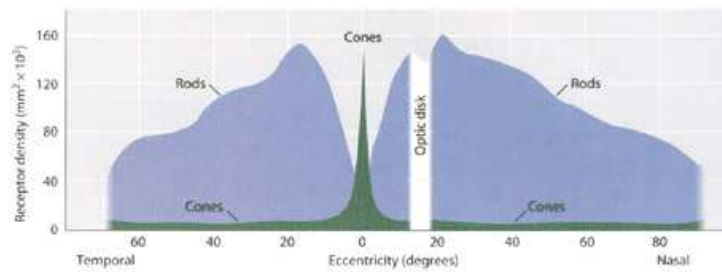


Figura 1.4: Relación de la distribución de los conos y bastones a lo largo de la retina [?]

Debido a ésto, el detalle de la porción de escenario que es capturada en la zona periférica es menor ya que el bastón, al ocupar más área por su tamaño, percibe una porción más grande del escenario aunque debe sintetizarla en una señal que debe ser transmitida a la zona cerebral. Por otro lado, la ventaja del cono, al percibir una menor porción del escenario, es que no tiene que sintetizar tanta información para enviarla al cerebro por lo que el nivel de detalle es mayor en los conos y por lo tanto en la zona foveal.

Como ya se dijo, una de las ventajas es la compresión de la información que es adquirida y enviada al cerebro para su análisis. El punto ahora es comprender por qué pese a tener un campo de visión de resolución no uniforme, se tiene la impresión de que es uniforme. La respuesta a ello es la manera en como el sistema de visión humano efectúa la visualización y el reconocimiento de formas.

Como primer punto, el hecho de que el sistema de visión humano tenga mayor detalle de información en el centro no implica que el resto de la zona retinal es inferior o tiene un funcionamiento menos relevante en el proceso de visión. Simplemente cada zona (foveal y periférica) es utilizada para diferentes tareas. La zona central de la retina o región foveal es utilizada para analizar, a nivel muy alto de detalle, información del escenario. Se pueden distinguir colores y al utilizar la información de la zona foveal obtenida con el otro ojo, es posible obtener información de profundidad de calidad. De aquí recibe el nombre de *visión foveal* y que se refiere al procesamiento de la información obtenida por la fovea.

Por otro lado, la *visión periférica* procesa toda la información obtenida a lo largo de la zona periférica, la cual es recolectada por los bastones distribuidos a lo largo de

esta zona. Esta es utilizada para procesar información de manera más cruda o con menor detalle. Es utilizada para tener un panorama global del escenario así como para distinguir la presencia de objetos más grandes. Tiene un reconocimiento muy limitado de colores y la información de profundidad que se puede obtener es muy pobre.

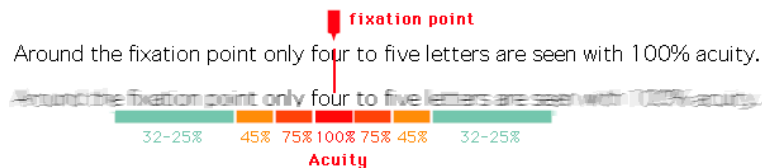


Figura 1.5: Ejemplo de procesamiento foveal y periférico [?]

Un ejemplo de estos tipos de visión o procesamiento de regiones en el ojo humano se muestra en la figura 1.5. Este ejemplo muestra el concepto de la fijación sobre un escenario y con este se explica por que se mueven los ojos de izquierda a derecha mientras se va leyendo un texto.

Como puede apreciarse en la figura, la primera línea de texto muestra como se cree que es percibida y esto es, con un campo de visión uniforme y por lo tanto con un porcentaje total de captura. Sin embargo, *fijando* la mirada en la posición señalada con la línea vertical, la cantidad de información que se percibe va decreciendo conforme la imagen se aleja del centro de fijación. La segunda línea de texto muestra la diferente resolución del escenario como en realidad es percibido por el ojo y que queda representado por un escenario borroso. Abajo de la línea, los números indican el porcentaje de información que es adquirido por el ojo el cual es no uniforme.

Al existir en la retina dos zonas con diferente resolución de recepción, entonces los módulos que procesan a esta información son diferentes entre sí también. Una pregunta se deriva de esto: ¿Cómo se integra esta información para llevar a cabo procesos como la identificación de objetos, fijar la atención, reconocer objetos, seguir objetos entre otros?

Actualmente esta pregunta no está completamente contestada y se encuentran muchas incógnitas aún sin resolver debido a la gran complejidad estructural del cerebro. Sin embargo, se sabe que hay una transmisión de pulsos y se conoce que ciertas actividades estimuladas por el sistema de visión provocan actividad cerebral en de-

terminadas zonas.

1.4.1. Fijación y movimientos sacádicos

Además de conocer qué partes del cerebro se activan durante las actividades de reconocimiento y seguimiento, existen algunas investigaciones que muestran cómo el sistema de visión es utilizado para desempeñar ciertas actividades. En este trabajo nos interesa en particular la actividad de seguimiento de objetos y para ello se deben considerar dos conceptos importantes en el proceso de visión que son la *fijación* y los *movimiento sacádicos* que tienen que ver con actividades específicas del movimiento del ojo en combinación con la información que se percibe en la retina.

De acuerdo a diversas investigaciones, el ojo humano puede asumir solamente dos estados: la *fijación* y la *saccade*, palabra francesa que significa salto, de aquí se deriva el término *movimiento sacádico* o movimiento de saltos.

La **fijación** se refiere al estado del ojo en el que éste concentra la atención sobre un objeto de interés y específicamente, dirigiendo a la zona foveal sobre la porción de campo de visión en la que se encuentra ubicado dicho objeto.

La **saccade o movimiento sacádico** un movimiento del ojo de gran velocidad y corta duración. Este movimiento es un salto instantáneo que el ojo produce para obtener información de manera rápida de diferentes zonas en el campo de visión donde se ha detectado la presencia de objetos en la zona periférica y cuya importancia puede ser relevante de acuerdo a un criterio determinado.

Se sabe que el 95 % del tiempo el ojo permanece en el estado de fijación o estado de atención sobre un objeto de interés. En este estado el ojo permanece relativamente estático o sujeto a movimientos suaves producidos para mantener dentro de la región foveal a un objeto que se encuentre desplazándose. Por otra parte, el estado de salto es utilizado con menor frecuencia para cambiar de objeto de interés sobre alguno que haya sido detectado en la zona periférica y sobre el cual se desea obtener mayor información. Para ello el movimiento saccádico dirigirá la zona foveal sobre la porción

del escenario en donde se encuentra el objeto detectado en la periferia [?], [?].

Otra situación que puede provocar un cambio al estado de movimiento sacádico es la pérdida parcial o total del objeto, ya sea por una oclusión o por un movimiento rápido de éste. Esto produce que el ojo dé saltos para buscar sobre el escenario posibles candidatos del objeto de interés perdido y fijar sobre de ellos nuevamente la atención del ojo y en específico, para ubicarlo dentro de la zona foveal nuevamente. Los saltos no son aleatorios y lo que se hace es dirigir la fijación hacia aquél objeto cuya forma y dimensiones sean semejantes al objeto que se estaba observando.

Finalmente, como todo sistema natural, los estados de fijación y de movimiento sacádico en el ojo no son perfectos y se encuentran en continuo entrenamiento a lo largo de la vida. El envejecimiento y el desgaste al que los ojos se ven sometidos diariamente produce un debilitamiento en ambos estados.

En edad temprana, el ojo puede ser entrenado para mejorar no sólo sus estados, sino también la percepción y procesamiento en la visión foveal y la visión periférica. Por ejemplo, se sabe que hasta la edad de siete años es posible comenzar a reconocer formas y objetos en la visión periférica. Lo mismo ocurre para procesar la información recibida por la zona foveal para reconocer un objeto de manera eficaz.

Uso de la visión foveal en la visión por computadora (seguimiento de objetos)

En el capítulo anterior se presentó y describió de manera breve en que consiste la visión foveal. En este capítulo se presentan los trabajos más representativos que han propuesto incorporar los fundamentos de la visión foveal en la visión por computadora para resolver una de las tareas de mayor importancia: el seguimiento de objetos.

2.1. El seguimiento de objetos

El seguimiento de objetos puede ser dividido en dos problemas principales los cuales se han atacado en diferentes formas. (1) El seguimiento de un sólo objeto y (2) el seguimiento de múltiples objetos. El primero consiste en localizar y seguir la trayectoria de un objeto elegido en el escenario y a lo largo de una secuencia de imágenes. El segundo implica localizar y seguir la trayectoria de diferentes objetos seleccionados en el escenario.

El seguimiento de un objeto en la escena parece un problema más simple que el de múltiples objetos sin embargo, no es menos difícil de resolver. Diversos factores complican el seguimiento volviéndolo arduo, sobre todo cuando existen restricciones de procesamiento en tiempo real. Estos factores pueden ser por ejemplo: la dinámica (el tipo de movimiento) del objeto, la estructura física del objeto (si tiene una forma geométrica o cambiante), oclusiones parciales o totales, el estado del fondo producido

por el estado de la cámara (fija o móvil), objetos adicionales en la escena, los cambios en la iluminación del escenario y el tamaño de la imagen a procesar.

Por otra parte, la problemática orientada a resolver la tarea de seguir múltiples objetos conlleva a aplicar algoritmos de seguimiento con una perspectiva diferente que se dificulta aún más tomando en cuenta los factores anteriores. Aunque muchos de los algoritmos desarrollados para la detección y el seguimiento son probados primero en el caso del seguimiento de un objeto y posteriormente, modificados para ser adaptados al caso de múltiples objetos.

Esto hace que el problema de seguir un sólo objeto no quede en el olvido, por el contrario, se siga tomando como un reto sobre el cuál se desarrolle ciencia básica para crear mejores métodos de localización, métodos para seguimiento de trayectorias, métodos para enfrentar las oclusiones o para enfrentar ambientes confusos como escenarios donde hay humo, neblina u objetos moviéndose continuamente. Así mismo se continúan desarrollando métodos para hacer posible esta tarea en menores cantidades de tiempo significando un procesamiento de más cuadros por segundo. Esto conducirá al desarrollo de sistemas más eficaces para el seguimiento de cuerpos que se desplazan a alta velocidad como aviones o proyectiles, por mencionar algunos.

La simple tarea de procesar el seguimiento de un objeto en una menor cantidad de tiempo implica todo un desafío para lo cual no basta contar con computadoras cada día más veloces o con más recursos, sino también con algoritmos optimizados para ejecutar en tiempos polinomiales de bajo orden y con un buen desempeño.

2.1.1. Regiones o ventanas de procesamiento

En este trabajo, el tema central es el problema del seguimiento de un objeto en tiempo real a lo largo de una secuencia de imágenes. El tiempo real es una cota que se puede fijar de acuerdo al tipo de aplicación, por ejemplo en varias de ellas el tiempo exigido es tal que se procesen 24 cuadros por segundo para un seguimiento exitoso. Generalmente la velocidad del objeto a seguir influye en el establecimiento de la cota mínima. Este punto es lo que vuelve aún más difícil el problema del seguimiento ya que, al estar sujetos a esta restricción de procesamiento, se deben elegir con cuidado los métodos para la localización y el mantenimiento de la trayectoria del objeto.

La mayoría de las técnicas que se utilizan para el seguimiento de objetos se ha

desarrollado para procesar imágenes obtenidas con sensores y cámaras digitales de resolución uniforme. Como se explicó en el capítulo 1, una cámara de resolución uniforme implica que el escenario es muestreado en la misma proporción por cada uno de los receptores que conforman al sensor. Sin embargo, el tiempo de procesamiento que es exigido por los métodos de seguimiento es costoso aún para las computadoras de hoy en día.

Por lo anterior, sólo una porción del campo de visión observado puede ser procesado y esto significa que de toda la imagen adquirida por el sensor, sólo una parte es utilizada para llevar a cabo el procesamiento para localizar al objeto. Esto es común al procesar imágenes de resolución uniforme ya que muchos de los algoritmos pueden tener órdenes polinomiales de grado tres ($O(n^3)$) o incluso superiores.

En diversos trabajos una forma sencilla de procesar una porción de la imagen consiste en definir lo que se le denomina una *ventana de búsqueda*. La ventana de búsqueda es una región rectangular que se define en la imagen adquirida para restringir el área de procesamiento; permitiendo así reducir el número de píxeles a analizar.

La ventana de búsqueda tiene algunos inconvenientes. El primero de ellos emerge cuando se define el tamaño, éste puede variar de aplicación en aplicación e incluso varía dependiendo del tamaño del objeto. Otro problema que puede surgir es que la velocidad de captura de la cámara sea menor a la velocidad con la que se desplaza el objeto a seguir y esto produzca que en la siguiente imagen capturada el objeto aparezca parcial o totalmente fuera de la ventana, provocando su pérdida o una detección errónea. Otro problema es el conocido como *problema de apertura* que ocurre cuando el objeto de interés aumenta o disminuye su tamaño ya sea, al acercarse o alejarse a la cámara o porque el sistema de visión es el que se aproxima o aleja del objeto.

2.2. La visión foveal en la tarea de seguimiento de objetos

La idea de utilizar la visión foveal en el problema de seguimiento de objetos está basada fundamentalmente en los estados de fijación y de saccade del ojo (Sección

1.4.1). Como premisa inicial se establece el no trabajar con un campo de resolución uniforme, sino con uno no uniforme que permita elegir el detalle de las regiones de trabajo. Para ello, se establece que la zona de trabajo o *región de procesamiento*, deba ser aquella con mayor resolución para aplicar de manera efectiva los algoritmos de localización, análogo al estado de fijación. Para el resto del escenario, la resolución debe cambiar bajo una regla propuesta de tal forma que la calidad de los datos disminuya conforme se aleje uno del centro. La segunda premisa de igual importancia y aún correspondiente al estado de fijación, consiste en mantener la alta resolución sobre el objeto de interés, incluso cuando éste se encuentre en movimiento. El estado de movimiento sacádico es aplicado cuando el objeto de interés se pierde o se desea cambiar de objeto de interés y para ello, es necesario realizar una exploración menos aguda sobre el resto del escenario hasta localizarlo.

La primer ventaja inmediata al hablar de trabajar con una resolución no uniforme es la reducción de información que se logra al muestrear el escenario, siguiendo la idea del muestreo que hacen los receptores del ojo. Esta reducción resulta conveniente al pensar en definir ahora una región o ventana foveal en vez de una ventana de búsqueda en donde se supone que el objeto de interés se encontrará. Otra ventaja es que al trabajar con la resolución no uniforme, las imágenes obtenidas, sobre las que se realiza el seguimiento, podrán ser más pequeñas. Esto hace factible el procesar la imagen con resolución no uniforme ¹ en vez de sólo una porción, por lo que aún cuando el objeto salga de la región foveal, su localización puede ser realizada en la periferia, ya que esta información está contenida en la imagen y no restringida o segmentada como en una región o ventana de búsqueda.

De forma natural, tomando las ideas anteriores, la visión foveal ofrece una solución atractiva para efectuar el seguimiento de objetos aunque, surgen algunos problemas y restricciones, como son:

1. Se debe pensar en cómo sustituir el modelo foveal o cómo adaptarlo al modelo uniforme o *Cartesiano* con el que trabajan las cámaras actuales. A este proceso de conversión o adaptación se le denomina *foveación*. ².

¹A la imagen con resolución no uniforme, obtenida mediante la aplicación de los principios de la visión foveal, se le conoce como imagen foveal

²Al modelo uniforme se le conoce como *Cartesiano* debido a que en una imagen digital, cualquier pixel puede ser referenciado por sus coordenadas (x, y) , indicando que éste está en la columna x y

2. El sistema de visión foveal humano implica que sólo un objeto puede ser colocado dentro de la región foveal por lo que, el seguimiento de múltiples objetos queda descartado. Esta restricción va sujeta a la premisa de mantener al objeto de interés dentro de la región foveal por lo que se tiene que pensar no sólo en un esquema de foveación, sino también en uno que permita cambiar la región foveal dentro de la escena, a lo que se le denomina *fovea móvil*.
3. Los métodos de seguimiento convencionales han sido diseñados para trabajar con imágenes de resolución uniforme y al adaptar de algún modo el modelo foveal, se deben desarrollar o adaptar estrategias que puedan trabajar con la resolución no uniforme.

Los tres puntos anteriores han significado todo un reto, el cual se ha ido atacando con diferentes propuestas. En las siguientes secciones se exponen los trabajos más representativos que han sugerido una respuesta a los tres planteamientos anteriores.

2.3. Primer modelo matemático: la transformada log polar

La aportación en la investigación de Schwartz no sólo fue la descripción del sistema de visión de los primates, sino también el modelo matemático con el cuál evidenció sus teorías. Este modelo aproxima la distribución de los fotorreceptores en la retina mediante una regla matemática sencilla. Primero, el modelo supone a los fotorreceptores como una malla de puntos divididos en sectores y anillos. En la intersección de cada sector con cada anillo, se encuentra posicionada el área central de un fotorreceptor. Otra manera de verlo es como un conjunto de anillos de diferentes radios que contienen un número igual de divisiones. En cada división hay un fotorreceptor y esto implica que en cada anillo existe un mismo número de fotorreceptores.

La distancia que separa a un anillo de otro es establecida por una proporción logarítmica que inicia a partir de la zona central. Esta proporción produce el efecto de una alta concentración de fotorreceptores y dicha concentración comienza a decrecer conforme los anillos se van separando del centro. La figura 2.1 muestra un ejemplo la fila y en la matriz de datos capturados.

de la malla de anillos y sectores con la que se modela la distribución. En la imagen a la derecha se muestra de manera más explícita, el centro de cada uno de los fotoreceptores distribuidos con base en estas reglas. En esta imagen se puede apreciar claramente la alta concentración de fotoreceptores que captarán mas detalle del campo de visión mientras que los demás fotoreceptores en la zona periférica comienzan a separarse más entre ellos por lo que tendrán que abarcar una mayor porción del escenario.

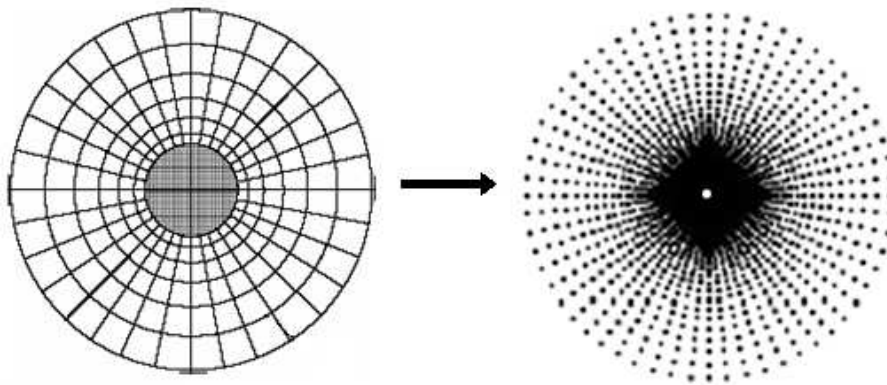


Figura 2.1: Distribución de los fotoreceptores de acuerdo a Schwartz

El modelo matemático consiste en caracterizar la distribución de cada fotoreceptor mediante dos datos: su distancia euclidiana con respecto del centro de la retina, la cual está determinada en proporciones logarítmicas, y el ángulo o los grados en los que se encuentra orientado con respecto del centro.

Si se colocara a la retina en un plano Cartesiano, con el centro de coordenadas localizado sobre el centro de la retina, cada fotoreceptor tendría una coordenada bidimensional (x, y) . Sin embargo, esta localización no es muy conveniente ya que lo esperado es agrupar en un sólo conjunto las coordenadas de cada uno de los fotoreceptores. El tamaño de este conjunto debe ser mucho menor que el total de coordenadas existentes en el plano mínimo necesario para mapear a cada uno de los receptores. Esto cobra sentido al tener la intención de almacenar los valores obtenidos por los receptores distribuidos de manera foveal en la retina y a su vez, contar con información de su posición espacial en el plano Cartesiano para posteriores análisis.

Para ejemplificar esto supóngase el conjunto:

$$\mathcal{FR} = \{(p_i, x_1, y_1), (p_i, x_2, y_2) \dots (p_i, x_n, y_n)\}$$

que contiene las coordenadas Cartesianas de los n fotoreceptores con p_i como el dato que almacena la información observada del escenario, y (x, y) las coordenadas Cartesianas de su ubicación en la retina. Una vez reunidos se plantea una pregunta de suma importancia: ¿qué criterio se toma para ordenarlos?

La *transformada Log Polar TLP* es un ordenamiento natural que surge del modelo de Schwartz al localizar a los fotoreceptores por sus coordenadas logaritmo-angulares, en vez de las Cartesianas. La transformación consiste en aplicar un mapeo que permite cambiar del plano Cartesiano a un plano conocido como plano *Log Polar* en el que se toman como ejes coordenados los valores angulares y logarítmicos radiales del fotoreceptor. De acuerdo al modelo de Schwartz, las coordenadas Cartesianas pueden ser transformadas en coordenadas log polares mediante las siguientes ecuaciones:

$$\xi = \ln(\sqrt{(x - x_0)^2 + (y - y_0)^2}) \quad (2.3.1)$$

$$\gamma = \tan^{-1} \left(\frac{y - y_0}{x - x_0} \right) \quad (2.3.2)$$

En estas ecuaciones, se aprecia que ambas coordenada ξ y γ tienen una singularidad cuando $x = x_0$ y $y = y_0$. Como consecuencia la TLP sólo puede muestrear pixeles cuyo radio sea de una distancia mayor a 0 con respecto al centro definido en la imagen. Esto produce un hoyo o falta de información en el centro de ésta.

La figura 2.2 muestra cómo se mapea una imagen del plano Cartesiano al plano Log Polar. Tanto en la ecuación 2.3.1, como en la ecuación 2.3.2, el centro de la retina es ubicado sobre una coordenada (x_0, y_0) lo cual será de mucha utilidad para definir una fovea móvil similar a una traslación de centro. Por otra parte, la transformación sugerida por estas ecuaciones muestra una forma sencilla y fácil de agrupar la información de los receptores en una estructura también matricial, que permite un manejo fácil y cómodo de la información percibida con sólo cambiar de plano. Nótese que se habla de una agrupación matricial siempre que las ecuaciones sean truncadas o

redondeadas a números enteros pues las coordenadas Log Polares pueden ser números reales.

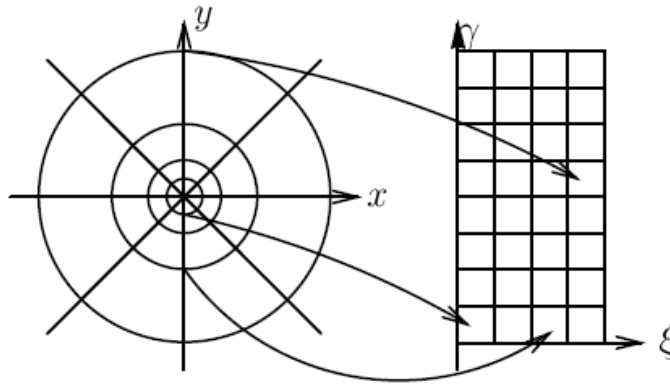


Figura 2.2: Mapeo del plano Cartesiano al plano Log Polar

Aunque la TLP otorga una agrupación simple de la información en el plano log polar, la mayor ventaja se tiene al aplicar este mapeo en el ámbito de las imágenes digitales que son a final de cuentas, un conjunto de píxeles o datos discretos agrupados en un plano discreto Cartesiano. Este plano que contiene a la imagen digital, se puede dividir en anillos y sectores a partir de un centro determinado tal y como lo establece la TLP. Una vez hecha la partición, cada zona delimitada por una intersección puede ser muestreada o promediada según convenga, para obtener un dato cromático que represente a la porción de la imagen en esa zona. Esa información será asignada al píxel con coordenadas Log Polares en el plano Log Polar y representa (o simula) al fotoreceptor en esa zona Cartesiana.

La figura 2.3 muestra un ejemplo de la superposición de la malla Log Polar sobre una imagen digital. Esta superposición ilustra cómo se lleva a cabo el mapeo para asignar los valores correspondientes de los píxeles en la imagen original, hacia los píxeles de la imagen resultante en el plano Log Polar. A esta imagen resultante se le denomina *imagen Log Polar*.

Los ejes en el plano Log Polar pueden ser asignados sin alguna preferencia en particular. Puede ser por ejemplo: el plano horizontal para denotar a la coordenada angular y el plano vertical para la coordenada radial-logarítmica, o viceversa. Por ejemplo, para la imagen foveal en la figura 2.3, los ejes han sido dispuestos de acuerdo

a la primera forma mencionada.

Como quiera que se haga la asignación, lo importante es señalar cual es el papel de estas imágenes en el proceso de seguimiento de objetos. Como primer paso, la imagen obtenida por una cámara CCD debe ser convertida a una imagen Log Polar, proceso que se puede realizar de manera muy sencilla mediante la aplicación de las fórmulas 2.3.1 y 2.3.2.

Una vez que se cuenta con la imagen Log Polar, el proceso de detección deberá ser realizado sobre ésta. Cuando el objeto se detecta, de alguna forma se deberá obtener una coordenada central que identifique la posición del objeto en la imagen y esto se hará en el plano Log Polar con las coordenadas (ξ, γ) , si el plano se ha dispuesto así. Con un manejo adecuado de las ecuaciones de la TLP y contando con las coordenadas Log Polares, es posible obtener la transformación inversa que entregue las coordenadas Cartesianas correspondientes a cada punto de la imagen digital original.

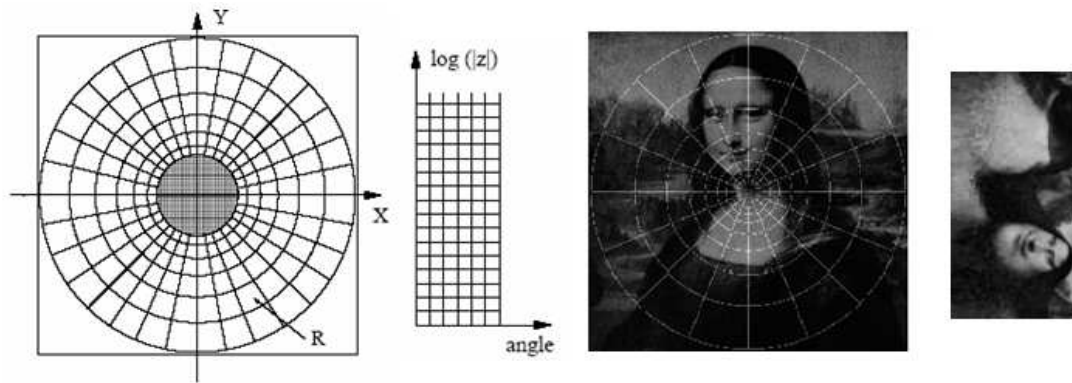


Figura 2.3: Aplicación de la Transformada Log Polar para mapear una imagen digital en una imagen Log Polar

2.3.1. Propiedades

Como puede apreciarse en la figura 2.3, la imagen foveal generada descompone el aspecto visual de la imagen original y pareciera que no hay modo de realizar una identificación de algún objeto de interés. Esta deformación de la imagen original ha representado un problema de gran interés y se han realizado muchos trabajos investigación para realizar el seguimiento utilizando imágenes Log Polares. En parte,

pese a la dificultad que presenta la forma visual de las imágenes, las imágenes Log Polares poseen algunas propiedades que las hacen atractivas para realizar seguimiento.

Para una mejor comprensión, primero asúmase que el plano Log Polar es un plano discreto - que es el resultado de un muestreo realizado sobre un plano Cartesiano discreto - y que corresponde a una imagen digital. El muestreo se realiza de acuerdo a las ecuaciones 2.3.1 y 2.3.2 de la TLP. A su vez, en el plano Log Polar se representa en el eje horizontal las distancias logarítmicas-radiales (ξ) y en el eje vertical a las distancias angulares (γ). Del mismo modo, considérese que un plano Cartesiano se posiciona sobre la imagen digital y se traslada el centro de coordenadas a la posición (x_0, y_0) .

División de la zona foveal y periférica

Para comprender esto se tiene que analizar la manera en que se lleva a cabo la transformación. De acuerdo a lo asumido anteriormente, el eje horizontal o cada columna de la matriz representa los radios logarítmicos de valor $0, 1, 2, \dots, \text{radio}_n$.

El eje vertical o cada fila representa a los ángulos posibles en los que los fotoreceptores pueden ser ubicados o bien los sectores. Estos serán dados por $k2\pi/N_{\text{Angulos}}$ donde $k = 1 \dots N_{\text{Angulos}}$.

Si se supone que hay un radio logarítmico de valor ρ_f a partir del cual, comienza a haber una separación angular entre los fotoreceptores o puntos de muestreo, entonces la información de muestreo, es decir de los fotoreceptores modelados como puntos, con radios mayores a 0, pero menores o iguales a ρ_f se encontrarán agrupados en la parte izquierda de la imagen tal y como lo muestra la figura 2.4.

Invarianza ante la rotación

Esta propiedad significa que si un pixel con coordenadas (x, y) (Cartesiano) en la imagen digital y coordenadas Log Polares (ξ, γ) , sufre una transformación de rotación, entonces este pixel sufrirá cambios únicamente en su coordenada Log Polar γ lo cual implica una traslación del pixel sobre el eje horizontal, en la imagen Log Polar.

Lo anterior se comprueba de manera formal del siguiente modo: para poder rotar un punto en el plano Cartesiano, hay que transformarlo primero en coordenadas radiales-angulares. Para un punto (x, y) en la imagen Cartesiana digital y el centro

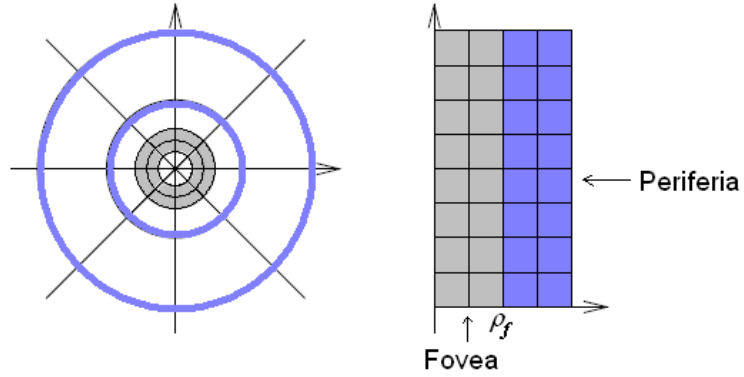


Figura 2.4: División de la zona foveal y la zona periférica en la imagen Log Polar

(x_0, y_0) se tienen las siguientes relaciones:

$$\rho = \sqrt{(x - x_0)^2 + (y - y_0)^2} \quad (2.3.3)$$

$$\theta = \tan^{-1} \left(\frac{y - y_0}{x - x_0} \right) \quad (2.3.4)$$

Donde las coordenadas Cartesianas pueden ser recuperadas a partir de estos datos y utilizando las ecuaciones correspondientes:

$$x = \rho \sin \theta \quad (2.3.5)$$

$$y = \rho \cos \theta \quad (2.3.6)$$

Ahora bien, supóngase que se desea rotar un punto un ángulo ϕ . Para ello, basta con sumar este ángulo al ángulo θ en las ecuaciones 2.3.6 y 2.3.5 lo que nos daría la nueva coordenada Cartesiana:

$$x' = \rho \sin(\theta + \phi) \quad (2.3.7)$$

$$y' = \rho \cos(\theta + \phi) \quad (2.3.8)$$

Transformando esta nueva posición a su equivalente en coordenada Log Polar se tiene que:

$$\rho = \sqrt{(x - x_0)^2 + (y - y_0)^2} = \sqrt{(x' - x_0)^2 + (y' - y_0)^2} \quad (2.3.9)$$

Ya que el punto sólo fue rotado y por lo tanto su radio con respecto al centro debe ser lo mismo. Por lo tanto se tiene que, las coordenadas Log Polares de la coordenada (x, y) son:

$$\xi = \ln \rho \quad (2.3.10)$$

$$\gamma = \tan^{-1} \theta \quad (2.3.11)$$

así de la nueva coordenada producto de la rotación:

$$\xi' = \ln \rho \quad (2.3.12)$$

$$\gamma' = \tan^{-1}(\theta + \phi) \quad (2.3.13)$$

donde $\xi' = \xi$.

De este modo, puede apreciarse que ambas coordenadas Cartesianas, la original (x, y) y la rotada (x', y') , comparten la misma coordenada Log Polar en el eje horizontal ξ , con diferente coordenada en el vertical para el punto rotado. Esto significa literalmente, que el pixel con coordenadas (ξ, γ) sufre una traslación en el eje vertical (ξ, γ') cuando este pixel es rotado en la imagen digital original. Este efecto puede apreciarse en la figura 2.5 en la imagen de la izquierda, que muestra el escalamiento de un conjunto de pixeles posicionados sobre uno de los anillos de muestreo en la imagen digital de coordenadas Cartesianas.

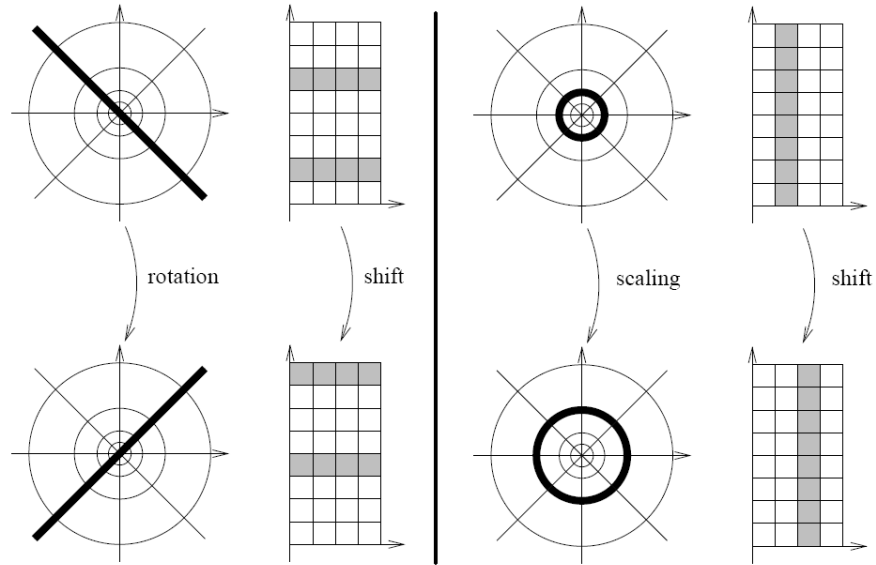


Figura 2.5: Invarianza de las imágenes Log Polares ante la rotación y el escalamiento

Invarianza ante el escalamiento.

El efecto del escalamiento de un pixel en la imagen digital tiene efectos similares al de la rotación cuando sus coordenadas Cartesianas son transformadas a coordenadas Log Polares. En este caso, el efecto es la traslación de la coordenada logarítmica-angular. Para comprobarlo se realiza un análisis similar al de la propiedad anterior.

Para escalar un pixel, basta con multiplicar las coordenadas Cartesianas (x, y) de él por un factor α . De este modo, de acuerdo a las ecuaciones 2.3.4, 2.3.3, 2.3.5 y 2.3.6, se tiene que la nueva coordenada será:

$$x' = \alpha x = \alpha \rho \sin(\theta) \quad (2.3.14)$$

$$y' = \alpha y = \alpha \rho \cos(\theta) \quad (2.3.15)$$

y utilizando el cambio de variable con $\rho' = \alpha \rho$, se tiene que:

$$x' = \rho' \sin(\theta) \quad (2.3.16)$$

$$y' = \rho' \cos(\theta) \quad (2.3.17)$$

donde se tiene que, para la coordenada Cartesiana (x', y') , las coordenadas radiales-angulares son (ρ', θ) . Calculando las coordenadas Log Polares respectivas para (x', y') se obtiene que:

$$\xi' = \ln \rho' \quad (2.3.18)$$

$$\gamma' = \tan^{-1} \theta \quad (2.3.19)$$

Como indican las ecuaciones 2.3.10 y 2.3.11, el punto o pixel con coordenadas (x, y) es transformado a las coordenadas Log Polares (ξ, γ) . El nuevo punto (x', y') obtenido por las ecuaciones 2.3.16 y 2.3.17, y que se refiere a la coordenada original escalada por una constante α , es transformado a las coordenadas Log Polares (ξ', γ') , donde $\gamma = \gamma'$.

Esto significa que al escalar un punto Cartesiano por una constante, esto producirá un efecto de recorrido en la componente logarítmica radial o sea sobre el eje horizontal, tal y como lo muestra la figura 2.5.

Para ejemplificar los efectos de estas dos últimas propiedades, la figura 2.6 muestra una imagen digital convencional que es rotada en diferentes ángulos y posteriormente escalada. Para esto, nótese el efecto que se produce en la imagen Log Polar y esto es, un recorrido en la filas y/o columnas de la imagen.

2.3.2. Ventajas

La primera e inmediata ventaja que ofrecen las imágenes Log Polares es la reducción de información obtenida por el muestreo. La cantidad de puntos que se muestrean se encuentra en dependencia del número de anillos y sectores en los que se divide la imagen Cartesiana digital. Para ello, diferentes propuestas se han sugerido concernientes a cómo realizar el cálculo de la TLP ya que, el modelo teórico es un modelo que trabaja con número reales y que debe ser adaptado al caso discreto por tratarse de imágenes digitales. En algunos trabajos como en [?] o en [?], se proponen diferentes modelos para llevar a cabo la foveación de la imagen digital convencional, siguiendo los principios establecidos por la TLP. Aunque pareciera trivial la implementación de

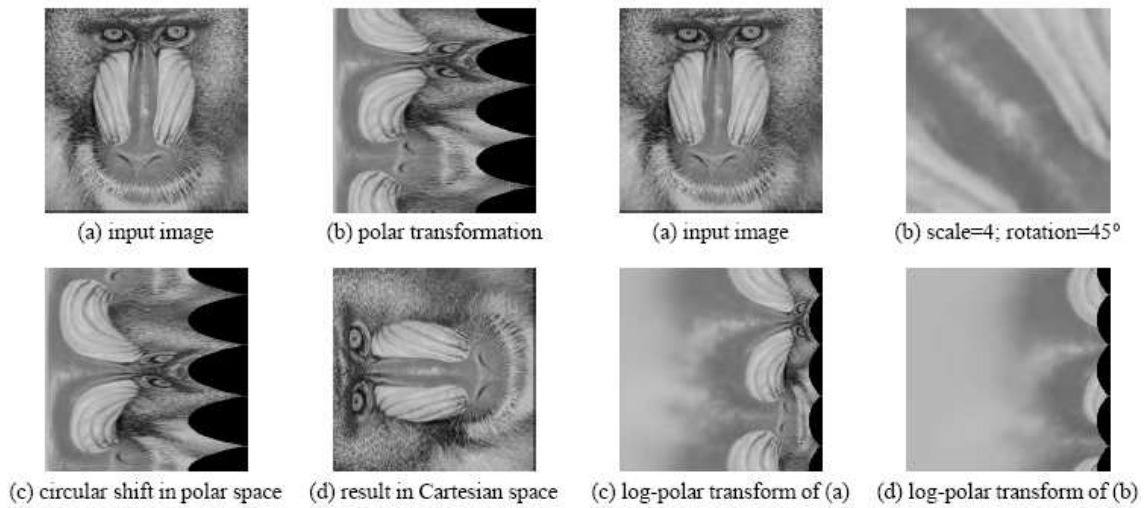


Figura 2.6: Efectos producidos sobre la imagen Log Polar al rotar y escalar la imagen digital que se muestrea

las ecuaciones 2.3.1 y 2.3.2 al caso discreto, existe un problema de computabilidad ya que al observar el mapeo sugerido por estas ecuaciones, inmediatamente resalta el hecho de que más de una coordenada Cartesiana será relacionada con una misma coordenada Log Polar. La estrategia de foveación que se seleccione debe de algún modo tomar en cuenta a todos estos puntos y sintetizarlos como un mismo dato, que se colocará en el pixel Log Polar correspondiente, o bien tomando al más representativo ya sea por posición, tono de color, o cualquier otro criterio. Así mismo, estas propuestas proponen utilizar una base logarítmica diferente al logaritmo natural para modificar el número de anillos y no sólo el número de sectores, como lo establece la TLP original.

Una ventaja unida a la definición matemática de la TLP es la sencillez del modelo mismo la cual no introduce operaciones complejas. Esto es conveniente para el control del tiempo de procesamiento ya que, adicionalmente al tiempo de captura y seguimiento, se debe sumar el tiempo utilizado por el proceso de foveación. Este modelo no debe consumir mucho tiempo y esto estará en dependencia del procedimiento que sea elegido para procesar la TLP. El modelo más simple, que es tomar una relación uno a uno, de las ecuaciones en la TLP puede ser calculado y almacenado en una tabla ya que, a menos que se redefina el número de anillos o sectores,

las coordenadas Log Polares siempre se mantienen igual, bastará con calcularlas una sola vez y utilizarlas en cada captura para realizar el proceso de foveación. La calidad de las imágenes resultantes estará en dependencia de las necesidades de la aplicación para la que sean utilizadas.

Dejando a un lado el problema anterior, la siguiente ventaja es la división por áreas de la zona foveal y la zona periférica en las imágenes Log Polares. Esta propiedad es la base para establecer que el objeto de interés se mantenga siempre sobre la zona foveal por dos motivos: (1) la zona foveal tiene mayor resolución o información muestreada con mayor detalle de la imagen original. (2) La traslación de la fovea puede computarse fácilmente ya sea por software, variando el centro (x_0, y_0) sobre la imagen digital, o por hardware que implica el mover y mantener el centro óptico de la cámara sobre el objeto de interés.

Si lo anterior se asegura, la localización e identificación del objeto de interés se tiene que realizar solamente sobre la región foveal puntualmente dividida por el radio ρ_f que indica la frontera entre la zona foveal y la periferia. Este valor puede ser calculado fácilmente como es sugerido en [?]. El realizar la localización sobre esta área restringida por ρ_f , conlleva a una reducción substancial de tiempo de procesamiento lo cual facilita un procesamiento mayor de cuadros por segundo.

Por otra parte, la invarianza en rotación y escalamiento resulta atractiva para resolver el problema del reconocimiento de un objeto de interés después de que éste ha sufrido una rotación (se puede pensar en un avión), o un escalamiento derivado, ya sea un acercamiento o alejamiento. Para ello, esta propiedad de invarianza puede explotarse ya que estos cambios implican un corrimiento en columna o fila en la imagen Log Polar lo que sugiere una búsqueda restringida a sólo procesar y empear filas o columnas. En [?] se ha sugerido un método para resolver este problema de identificación y correspondencia mediante el uso de las imágenes Log Polares utilizando la Transformada de Fourier. Este trabajo se extiende incluso a la detección y seguimiento de un objeto aplicando la Transformada de Fourier al mapa Log Polar cuyas coordenadas logarítmicas-polares se pueden manipular algebraicamente con la transformada, aprovechando que también trabaja con números complejos. Así, el trabajo consistió en aplicar la Transformada de Fourier a las imágenes Log Polares y realizar una detección del objeto de interés en el espacio de frecuencias. Si el objeto se rota o cambia de tamaño, la invarianza a la rotación y el escalamiento en la TLP

ayudará a que estos cambios se vean como un desfase de señal en el espacio de frecuencias. La búsqueda o correlación se reduce a una multiplicación de funciones en este espacio. El gran inconveniente de esta propuesta es que el cálculo de la Transformada de Fourier puede ser un proceso igual de costoso que la foveación y de implementación costosa en hardware. Sin duda, existen restricciones para llevar a cabo esta solución para sistema en tiempo real y para escenarios exteriores en donde las condiciones ya no son ideales.

2.3.3. Desventajas

La desventaja más grande de las imágenes Log Polares está relacionada con su no linealidad. Esto significa visualmente que la imagen capturada en representación Cartesiana simple, al ser transformada a Log Polar, pierde su uniformidad y además sufre transformaciones abruptas de su forma por lo que, visualmente, resulta difícil reconocer el aspecto original del objeto.

De manera formal, el cambio de coordenadas provoca que se pierda la relación de vecindad entre pixeles que existe en el plano Cartesiano. Esto es, dos pixeles adyacentes en el plano Cartesiano pueden ya no serlo en el plano Log Polar. Por lo tanto, aunque el objeto de interés se localice sobre la zona foveal (región izquierda de la imagen Log Polar), su identificación exacta se vuelve ardua al encontrarse dispersada la información de los pixeles que lo conforman en la imagen original. Por otra parte y sobre el mismo punto, muchos de los métodos de localización e identificación del objeto, son métodos lineales diseñados para trabajar con las propiedades de vecindad de los pixeles. Su aplicación se vuelve ineficiente y representa todo un reto el adaptarlos en las imágenes foveales. Además como ya se ha mencionado, a este problema se suma el que estas imágenes no son aptas para realizar el seguimiento de múltiples objetos debido a que sólo una región en la imagen posee alta definición en la información y sólo relacionada con el centro de la imagen.

En cuanto a las propiedades de invarianza ante la rotación y el escalamiento, éstas sólo se mantiene si el objeto de interés sufre las transformaciones estando posicionado exactamente sobre el centro de la fovea lo cual no ocurre frecuentemente.

2.3.4. Aplicaciones

Pese al gran problema de la no linealidad de las imágenes Log Polares, diversos trabajos han aprovechado las ventajas ofrecidas para resolver algunas tareas relacionadas con la visión por computadora y a problemas específicos en el seguimiento de objetos.

Las imágenes Log Polares dificultan la localización de un objeto al interior de ellas, pero evaluar la semejanza entre un par de imágenes log polares es posible con cualquier medida de similitud. Esto fue aprovechado en el campo de aplicación de los *sistemas de visión estéreo*.

Un sistema de visión estéreo utiliza dos o más imágenes capturadas con diferentes cámaras de manera simultánea y colocadas en distintas posiciones sobre el escenario. El escenario que se observa es el mismo pero de diferentes ángulos; y las distancias a las que se encuentran colocadas las cámaras son conocidas. A las imágenes capturadas por esta configuración de cámaras se les denomina *imágenes estéreo*.

Muchas de las aplicaciones estéreo utilizan sólo dos cámaras colocadas a una distancia y ángulos conocidos, lo que se conoce como sistema de cámaras calibrado. Dado un par de imágenes estéreo, el problema consiste en obtener un mapa de profundidades de los objetos contenidos en la escena, implica un problema de correspondencia de puntos en las imágenes.

La obtención del mapa de profundidad implica dos problemas: (1) el problema de correspondencia entre las imágenes estéreo y (2) la reconstrucción del modelo 3D de la escena a partir de este mapa de disparidades. El primer problema se puede resolver encontrando la correspondencia entre los píxeles entre las imágenes estéreo. Este problema es atacado con algoritmos que encuentran similitudes entre las imágenes y ubican la mejor correspondencia de los puntos en ambas. Una vez conociendo la correspondencia de cada punto, el cálculo del mapa de disparidad es simple. Sin embargo, el proceso de similitud consume tiempos altos de procesamiento los cuales se incrementan si el tamaño de las imágenes aumenta.

El papel de las imágenes Log Polares en la visión estéreo consistió en obtener imágenes Log Polares del par de imágenes estéreo, y llevar a cabo el proceso de correspondencia entre el par de imágenes Log Polares. Este proceso es más rápido debido al menor tamaño de las imágenes y puede optimizarse aún mas gracias a los

cambios de resolución que diferenciarán en gran medida ciertas áreas de la imagen. Así, el mapa de disparidad Log Polar obtenido puede ser utilizado para recuperar un mapa de disparidad Cartesiano, utilizando la transformada inversa de la TLP. Este mapa de disparidades tendrá un efecto foveal con alta precisión de la profundidad en el centro o región donde se haya muestreado la fovea, y menor en la periferia.

Esta idea fue utilizada para implementar el control de un robot para la tarea de seguimiento de un objeto en un escenario interior. La idea básica consiste en dotar al robot o al sistema de un par estéreo de cámaras CCD. El sistema se posiciona sobre el objeto de interés, se obtiene el mapa de disparidad mediante el proceso descrito anteriormente, el cual consumirá menos tiempo, y como el objeto de interés se encuentra sobre la región foveal, entonces el mapa de disparidad obtenido tendrá información detallada de su forma tridimensional. Esto servirá como modelo 3D para ubicarlo en un instante de tiempo posterior. Para hacer eso, se lleva a cabo un proceso de localización utilizando los mapas de disparidad obtenidos por el proceso anteriormente descrito, y utilizando el modelo 3D obtenido [?].

Lo anterior, ha servido como base para diversos trabajos de investigación que han estudiado las propiedades matemáticas de las imágenes Log Polares para llevar a cabo el proceso de similitud entre dos imágenes foveales estéreo. Por ejemplo en el trabajo de Oshiro en [?] y [?], se estudia la influencia del campo de visión de las cámaras estéreo y el efecto que producirán en las imágenes Log Polares lo cual puede ayudar a optimizar el proceso de correspondencia. Esto es aplicado para la tarea de seguimiento binocular de un objeto utilizando un sistema de vigilancia de dos cámaras que realiza el seguimiento tomando como modelo el modelo 3D del objeto, identificándolo y siguiéndolo a lo largo del tiempo.

Otro ejemplo es el trabajo de Bernardio en donde este tipo de sistema estéreo Log Polar fue utilizado para el control de un robot móvil que, utilizando las imágenes Log Polares como medio de visión, decide como desplazarse sobre su entorno, identificando y evadiendo obstáculos [?, ?, ?, ?]. Esta propuesta es limitada para aplicaciones estereo donde se requiere extraer un mapa de disparidad para mediciones de alta precisión. De cualquier modo, las aplicaciones derivadas son diversas como seguimiento de personas, evitación de obstáculos en navegación, exploración, entre otros.

Una tarea involucrada con la visión por computadora y que puede ser utilizada como complemento para el seguimiento de objetos, es la estimación del flujo óptico.

Este problema exige que la cámara que captura las imágenes se mantenga fija. La idea aquí consistió en obtener la secuencia de imágenes foveales correspondientes y calcular el flujo óptico en estas. Este trabajo fue realizado por Kruger en su tesis de doctorado [?]. Nuevamente, las ventajas de abordar este tipo de problemas sobre las imágenes Log Polares son la reducción de datos a procesar y que la definición matemática de la TLP ayuda a la estimación del flujo. Desafortunadamente, el flujo óptico puede ser utilizado para estimar la trayectoria del objeto y conocer de manera probable la posición del mismo en la siguiente escena, no así con la forma del objeto, lo cual debilita el seguimiento cuando existen oclusiones.

El problema de la no linealidad como ya se mencionó, restringe en gran proporción el desarrollo de aplicaciones para seguimiento de objetos y en varias de estas aplicaciones no es adecuado o resulta difícil el incorporar dos cámaras calibradas para obtener imágenes estéreo. Por otra parte, el método utilizado para las imágenes estéreo funciona sólo para escenarios en interiores lo cual limita aún más el campo de aplicación. No obstante han habido algunos intentos como en [?], donde se propone el seguimiento de un objeto en una escena en espacios cerrados o interiores. Este sistema tiene más un toque de un sistema de vigilancia que comienza a seguir al primer objeto que entra a la escena. Las debilidades de esta propuesta es que precisamente, por no poder realizar una localización o identificación del objeto en la imagen Log Polar, se realiza un análisis de la información de la dinámica del objeto y su tamaño. Desafortunadamente, el tipo de objetos que este sistema puede seguir son objetos con formas geométricas muy definidas. Además, se asume un fondo fijo y las oclusiones o presencia de más objetos puede confundir al algoritmo de seguimiento implementado.

Finalmente, existen dos propuestas que atacan el problema del seguimiento de un objeto mediante el uso de algoritmos de clasificación y aprendizaje. Tal es el caso de la discriminación y seguimiento de rostros en la escena usando imágenes Log Polares, realizado en [?] y [?]. En ambos trabajos se argumenta que el seguimiento puede ser hecho por un sistema de detección previamente entrenado con un conjunto de patrones del objeto de interés. Estos patrones se obtienen como una subimagen en la imagen log polar en donde se encuentra ubicado el objeto de interés. A estos datos se les extraen los componentes principales PCA y se usan para entrenar un clasificador. La idea resulta interesante, pero desafortunadamente no es muy viable para sistemas de seguimiento que no posean un modelo previo del objeto a seguir.

De manera alterna, las imágenes Log Polares se han utilizado para otras aplicaciones en visión como es, la identificación de características de escenarios interiores [?]. Esto resulta útil para la tarea de exploración en robots que se deben desplazar en un escenario cerrado y que deben identificar características de su entorno para así, construir mapas del mismo. Sobre esta misma línea, las imágenes Log Polares se han utilizado también para identificar esquinas en escenario interiores [?], lo que puede incluir el reconocimiento de puertas o ventanas.

En otros trabajos y saliendo un poco del área de seguimiento, en virtud de sus propiedades logarítmicas-radiales y angulares, las imágenes Log Polares se han utilizado para ocultar información en imágenes en el área de esteganografía. De manera más explícita, en aplicaciones donde se colocan marcas de agua en imágenes digitales, problema que se atacó con el uso de las imágenes Log Polares [?].

Como se ha presentado, la TLP como proceso de foveación ha inspirado el desarrollo de diversos trabajos dentro de la visión por computadora. Muchos de ellos relacionados con el seguimiento de objetos. Y aunque la TLP corresponde tal vez al modelo más formal de la visión foveal, sus limitaciones no han permitido desarrollar sistemas y aplicaciones para resolver problemas más complejos en cuanto al seguimiento de objetos compete.

Algunas propuestas diferentes han surgido para llevar a cabo el proceso de *foveación* con la idea de eliminar el problema de la no linealidad y la división espacial de los objetos en la imagen. En las siguientes secciones se presentan otros modelos que han intentado dar una mayor flexibilidad al proceso de foveación y por ende, brindar mayor facilidad para el desarrollo de algoritmos de seguimiento, incluso con la idea de incorporar los métodos convencionales que se usan en imágenes Cartesianas.

2.4. Geometrías cartesianas exponenciales

Originalmente propuesto por Bandera en [?], las Geometrías Cartesianas Exponenciales **GCE** se basan en la construcción piramidal de imágenes para el proceso jerárquico de búsqueda de un objeto de interés.

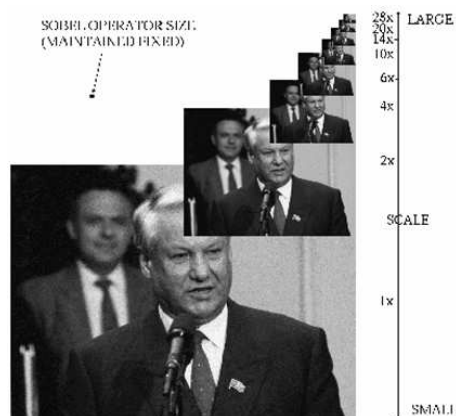


Figura 2.7: Construcción de una pirámide de imágenes para un análisis jerárquico

2.4.1. Análisis piramidal de imágenes

El modelo jerárquico multiresolución convencional consiste en producir un conjunto de imágenes a escala de la imagen original. Los cambios de escala en cada una de estas imágenes producen una vista del mismo campo de visión capturado por la cámara, pero a diferentes tamaños y por ende con diferentes resoluciones. Las imágenes son colocadas en un nivel jerárquico comenzando con la imagen de menor tamaño, a la imagen de mayor tamaño que, generalmente corresponde al tamaño de la imagen original. La figura 2.7 muestra un ejemplo de la construcción de una pirámide de imágenes con diferentes tamaños y resoluciones.

El método para escalar las imágenes puede ser cualquiera de los algoritmos tradicionales para escalamiento de imágenes, que puede ser desde un muestreo 1 de cada k píxeles, hasta un escalamiento utilizando Wavelets [?].

El proceso de seguimiento en un conjunto piramidal de imágenes, se ejecuta de la siguiente manera: primero se obtiene un modelo del objeto de interés en su tamaño original. Este modelo también es escalado a los diferentes tamaños correspondientes a los aplicados para producir la pirámide de imágenes. Posteriormente, se realiza un proceso de búsqueda iniciando en la primer imagen de la pirámide o bien, la de menor tamaño y resolución. En esta imagen se identifica y localiza al objeto de interés. Una vez obtenida su localización, se pasa al segundo nivel o la imagen son el segundo menor tamaño. En esta ocasión, la búsqueda no se realizará en toda la imagen, sino

en una vecindad determinada alrededor del punto de localización identificado en la búsqueda anterior. Una vez localizado, se pasará al siguiente nivel siguiendo la misma idea de manera sucesiva hasta llegar a la imagen con tamaño y resolución iguales a la original. Este proceso jerárquico tiene como objetivo realizar una identificación más veloz de zonas de interés donde el proceso de localización se va refinando conforme se cambian los niveles de resolución. Este procedimiento puede tener el inconveniente de ser sensible a la forma del objeto el cual, puede ser difícil de detectar si el escenario tiene bajo contraste con respecto al objeto de interés, lo que puede producir una búsqueda en casi toda la imagen en cada nivel de resolución. Esto implicará más tiempo de procesamiento del que se pretende economizar.

2.4.2. El polígono foveal

La diferencia de las GCE con el análisis piramidal de imágenes piramidal radica en el modo de escalar la imagen original para producir los diferentes niveles de la pirámide y que todas las imágenes que conforman a la pirámide tienen el mismo tamaño, más no la misma resolución ni el mismo campo de visión.

La idea para llevar a cabo el escalamiento consiste en dividir a la imagen original en anillos cuadrados de radio igual a una potencia de 2. El radio menor fijará el tamaño de todas las imágenes de la pirámide. Cada radio a su vez, determinará el campo de visión o el radio máximo del campo de visión que se visualizará en cada imagen de la pirámide. El radio menor también será la imagen de la pirámide con resolución igual a la resolución de la imagen original y por lo tanto será la imagen que se coloque en el último nivel de la pirámide. Para producir las imágenes restantes, se llevará a cabo un proceso de promediado en potencias de 2. Esto es, cada radio determinará el tamaño del área en píxeles que se promediará para constituir un píxel cuya información será colocada en la imagen. Por ejemplo, suponiendo que la imagen original sea una imagen de 16x16 píxeles y que ésta se divida en 4 radios de tamaño: $2^1 = 2$, $2^2 = 4$, $2^3 = 8$ y $2^4 = 16$, entonces la imagen en la base de la pirámide tendrá un área de 4x4 píxeles, al igual que el resto de las imágenes. Pero el campo de visión de las restantes estará determinado por los radios 4, 8 y 16. Para construir la imagen que le sigue a la base, se llevará a cabo un promediado de píxeles en una área igual a 2x2 píxeles. Para construir la tercer imagen, el área de promediado será de 4x4 y, para la

última imagen, el área total será de 8×8 píxeles. Esta imagen que se encontrará en la punta de la pirámide, tendrá el campo de visión completo de la imagen original con una resolución en potencia de 2. A esta forma de estructurar los radios y las áreas para promediar los píxeles y construir así, las imágenes de la pirámide, se le determina polígono foveal.

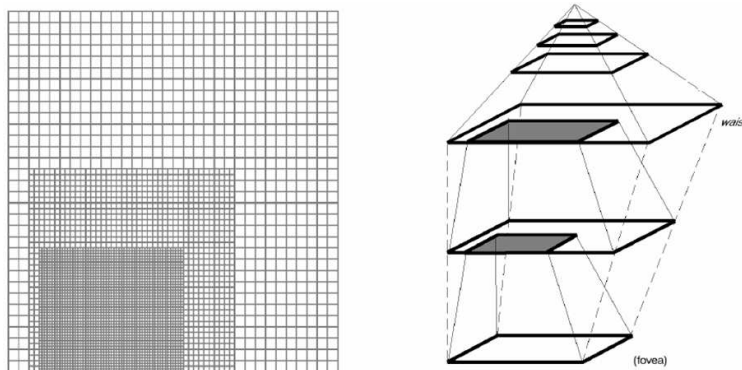


Figura 2.8: Polígono foveal y la construcción de la pirámide bajo el concepto de las Geometrías Cartesianas Exponenciales

En analogía con los principios de la visión foveal y bajo este esquema, una imagen digital es dividida en anillos cuadrados. La imagen de la base en la pirámide corresponde al área o región foveal con alta resolución. Las siguientes imágenes representan diferentes niveles del campo de visión y diferentes resoluciones siendo el último, el de mayor campo de visión con la menor resolución. Estas imágenes proporcionan un acceso a diferentes resoluciones de la zona periférica alrededor de la región foveal. Por otra parte, los radios o anillos son colocados a una *distancia exponencial* en vez de una logarítmica en analogía nuevamente con el concepto foveal aunque en este caso, se modela a los fotorreceptores como un conjunto de capas de fotorreceptores que contienen información a diferentes resoluciones y con diferentes campos de visión. La figura 2.8 muestra la división que realiza el polígono foveal sobre el plano y como se construye la pirámide de imágenes, en esta figura se aprovecha también para mostrar que el polígono foveal puede cambiar el centro de la región foveal para obtener la propiedad de la fovea móvil. La figura 2.9 muestra un ejemplo en concreto de cómo se descompone una imagen digital en la pirámide de imágenes del mismo tamaño usando el polígono foveal.



Figura 2.9: Ejemplo de la construcción de la pirámide multiresolución usando el polígono foveal sobre una imagen digital

2.4.3. Ventajas

La ventaja inmediata de las GCE es la linealidad de las imágenes que conforman a la pirámide. Esto permite que los métodos diseñados para trabajar con las imágenes Cartesianas puedan ser aplicados sin restricciones. Por otra parte, la búsqueda jerárquica se puede realizar dependiendo del tipo de aplicación.

En el caso del seguimiento de objetos la región foveal, representada por la base de la pirámide, funge como la primer imagen a analizar ya que se desea contar con alta resolución del objeto de interés. Si existiera la necesidad de analizar la periferia, cualquiera de los siguientes niveles de la pirámide puede ser utilizado dependiendo del nivel de detalle.

En cuanto al diseño mismo de la pirámide, el promediado por potencias de dos es un método muy simple de programar e incluso optimizar con técnicas de programación dinámica para la construcción de las imágenes en cada nivel. Incluso, su programación se puede llevar a nivel de hardware [?] ya sea en prototipado sobre FPGA, o en diseño de hardware en específico.

El polígono foveal por su parte, brinda versatilidad para mover el centro de la fovea y contar así con la propiedad de la fovea móvil. Lo mismo aplica en cuanto al cambio de los tamaños correspondiente a la región foveal y el número de anillos.

2.4.4. Desventajas

A pesar de las facilidades de computabilidad y reconfiguración, la generación de la pirámide representa un tiempo de cómputo que puede ser significativo para las aplicaciones en tiempo real. Por lo tanto, para cubrir esa restricción, es de suma

importancia que esta propuesta pueda ser implementada en hardware.

En cuanto a la identificación y la localización del objeto de interés, existe la inconveniencia de los diferentes radios de campo de visión que, obligan a la búsqueda del mismo en los siguientes niveles de la pirámide si es que el objeto no fue hallado en el nivel actual, mismo que puede incrementar el tiempo de procesamiento posiblemente no necesario en la búsqueda en una pirámide jerárquica convencional [?]. En esta misma línea, si el algoritmo de identificación del objeto no encuentra de manera precisa al objeto de interés en el nivel actual, entonces tendrá que pasar al siguiente nivel de la pirámide y si el objeto es identificado, entonces su localización tendrá cierto margen de error al tratarse de imágenes promediadas y con resolución variable. Este problema se incrementa conforme se avanza en los niveles de la pirámide.

2.4.5. Aplicaciones

Arrebola y Camacho han presentado algunos trabajos donde se utilizan las GCE. Por ejemplo, en [?] se presenta un sistema de visión activa foveal que analiza un escenario interior, segmenta y extrae el fondo del escenario, así como las características de los objetos presentes en él. De igual manera, en [?] se presenta un sistema de navegación autónoma para un robot móvil. En [?] se propone una metodología para obtener mapas de disparidad utilizando esta geometría. El inconveniente que tiene es que se procesa un cuadro por segundo por lo que queda limitada para su uso en tiempo real, aunque resulta efectiva para aplicaciones de medición 3D con alta precisión.

La cantidad de trabajos referentes a esta propuesta siguen líneas similares a estas y parte de ello se debe a las limitantes que producen las desventajas descritas anteriormente.

2.5. La transformada wavelet foveal

Propuesta por Wei en [?], la Transformada Wavelet Foveal **TWF** también se basa en la idea de la multiresolución sólo que, para el proceso de escalamiento se hace uso de la Transformada Wavelet con la adición de algunos filtros pasa banda y pasa baja para comprimir información en ciertas zonas de la imagen. Por la definición de los conceptos foveales, esta compresión se realiza en las zonas correspondientes a la



Figura 2.10: Aplicación de la Transformada Wavelet Foveal, a la izquierda imagen original, al centro descomposición piramidal utilizando la Transformada Wavelet con los filtros y a la derecha, reconstrucción de la imagen con el efecto foveal

periferia del campo de visión mientras que el centro permanece intacto.

Además de la descomposición piramidal, este trabajo integra en una sola imagen a las diferentes imágenes obtenidas por la aplicación de la transformada. Esta integración produce una imagen con un aspecto visual similar al campo de visión foveal. La figura 2.10 muestra un ejemplo de la descomposición y la imagen de aspecto foveal.

Esta transformada tiene varios méritos, el primero de ellos es que con ella se pueden producir imágenes foveales muy similares a las imágenes foveales naturales. Otra ventaja es la preservación de la linealidad de la imagen foveal generada e incluso, otra más significativa, es que la Transformada Wavelet permite trabajar en el espacio de las frecuencias temporales. Esto puede explotarse ya que algunos métodos de identificación o de búsqueda de similaridad son procesos que implican generalmente operaciones de convolución en el plano Cartesiano, pero que se simplifican a una multiplicación simple en el plano de las frecuencias.

Desafortunadamente, la principal desventaja de este trabajo radica en que la Transformada Wavelet implica un proceso complejo de computación, lo que la descarta inmediatamente para aplicaciones de tiempo real y por lo tanto para el seguimiento de objetos bajo estas circunstancias. Actualmente existen grupos trabajando en como implementar esta propuesta a nivel de hardware.

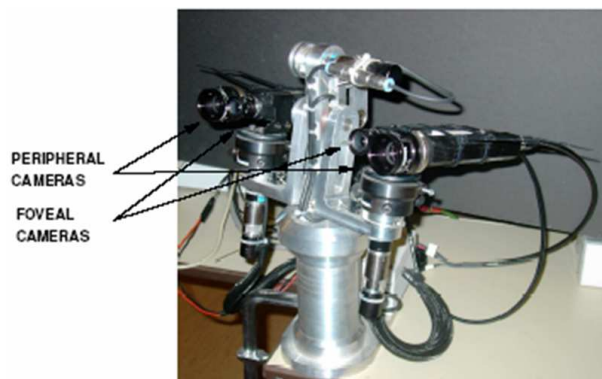


Figura 2.11: Sistema estéreo de 4 cámaras, 2 cámaras periféricas y 2 foveales

2.6. Vision foveal y periférica mediante el uso de cámaras de diferente resolución

Este trabajo fue propuesto en [?] y más que una solución para la tarea de seguimiento de objetos, fue un trabajo realizado para resolver el problema de la retroalimentación visual utilizada en robótica para el control. Puede ser utilizado para resolver problemas específicos dentro de las tareas de reconocimiento, seguimiento y estimación de posición de objetos.

La propuesta consiste en utilizar un sistema estéreo como el que se explicó en la sección 2.3.4, sólo que el sistema cuenta con dos pares estéreo de cámaras. Cada par tiene un campo de visión diferente donde uno es mas reducido que el otro. El par de cámaras con menor campo de visión tiene un paso de acercamiento mayor que el otro par. Esto permite observar con mayor detalle un objeto o región específica en el escenario. El otro par tiene un campo de visión más amplio y esto implica un menor detalle de los objetos en el escenario. La figura 2.11 muestra este sistema de cámaras.

Las imágenes obtenidas con las cámaras de menor campo de visión son utilizadas como regiones foveales, mientras que las otras son utilizadas como regiones periféricas.

El objetivo es identificar un objeto en el escenario utilizando las cámaras periféricas y, una vez localizado, utilizar las cámaras foveales para extraer detalles finos del mismo y hacer posible su seguimiento si es necesario en combinación con los modelos 2D y 3D que pueden ser extraídos gracias al sistema estéreo.

Desafortunadamente, el uso de un sistema de visión estéreo implica el cálculo de mapas de disparidad que, como se ha explicado ya, requiere un tiempo de procesamiento alto. Por otra parte, pese a la linealidad de las imágenes, el procesamiento implicado no cambia con respecto al que se realiza con las imágenes obtenidas por una cámara convencional esto es, este trabajo no está orientado para la reducción de información, sino como un trabajo que utiliza el concepto de visión foveal para organizar y ordenar el control de un sistema de visión activo. Para aplicaciones de seguimiento en exteriores o sistemas de tiempo real, este sistema puede ser limitado.

Algoritmos para el seguimiento

El seguimiento es una tarea importante dentro de los sistemas de vigilancia. Y aunque se han realizado numerosos esfuerzos para resolver esta tarea, aún no existe un método general de seguimiento que pueda aplicarse de manera efectiva a todos los escenarios posibles. El tiempo de respuesta del sistema, el ambiente de observación (interiores, exteriores) y el tipo de objetos a observar, son algunas de las restricciones que determinan la selección de los diferentes algoritmos utilizados para realizar un seguimiento aceptable.

En este capítulo se presentan los algoritmos que se adaptaron y eligieron para realizar la tarea del seguimiento basado en las condiciones del escenario donde se presenta el objeto de interés. La primer condición para la selección fue que estos algoritmos deberían poder ser aplicados sobre las imágenes foveales generadas con el método propuesto en este trabajo.

Otro punto importante a mencionar es que el algoritmo de seguimiento que se eligió es un algoritmo básico con el objetivo de probar que el seguimiento puede realizarse utilizando las imágenes foveales propuestas. Los algoritmos elegidos son bien conocidos en la literatura y se ha mostrado su efectividad en más de una publicación así como para las condiciones del escenario que se eligieron para trabajar.

3.1. Especificaciones del escenario

Como se mencionó anteriormente el uso de las imágenes foveales, generadas por las diferentes técnicas mencionadas en el capítulo 2, ha tenido una aplicación limitada tanto para el tipo de objetos a seguir como en el tipo de escenarios, la mayoría de

estos interiores.

De cierto modo, el seguimiento en escenarios interiores resulta menos complejo en virtud de que las condiciones del ambiente puede ser controladas como la iluminación, o bien puede extraerse un conocimiento previo como los colores del fondo, puntos donde hay esquinas u obstáculos. El seguimiento en escenarios exteriores representa un mayor reto al no tener un control completo de las variables mencionadas anteriormente. En este trabajo diversas secuencias con escenarios tanto exteriores como interiores fueron utilizadas para realizar el seguimiento.

La dinámica del fondo también es importante para decidir sobre la elección de los algoritmos de seguimiento. La dinámica del fondo depende generalmente de la posición de la cámara, la cual puede permanecer estática o en movimiento. Si la cámara permanece estática el fondo puede definirse como todo aquello que permanece sin movimiento a lo largo de la secuencia de imágenes observadas y/o capturadas por la cámara. Aún con la cámara fija, podrían haber objetos de este fondo que gradualmente pudiesen cambiar de posición, pero de ser así, estos objetos pasarían a formar parte de los objetos en movimiento de la escena. Del mismo modo, podrían haber objetos cuya dinámica pudiera disminuir y entonces permanecer en un estado cuasi-estático el suficiente tiempo para formar parte del fondo, de acuerdo a como éste se ha definido anteriormente.

En el caso de la cámara en movimiento, la definición de fondo introducida anteriormente no es de mucha utilidad ya que, al mover o desplazar la cámara, se pierde la noción de movimiento en la imagen capturada debido a que el movimiento se ha producido por un objeto que realmente se esté moviendo o bien, que un objeto se ha desplazado en la imagen a causa de que la cámara ha cambiado de posición.

Debido a que una cámara en movimiento o el fondo dinámico, producido por este u otro efecto, pueden introducir una mayor complejidad al problema de seguimiento, este trabajo se restringe al problema de seguimiento sobre escenarios con fondo fijo, el cual también es un problema de interés en diversos sistemas de visión.

3.2. Detección de movimiento

En un sistema de vigilancia, la detección de movimiento puede ser una etapa inicial de pre-procesamiento que indique probables blancos de interés. La selección del blanco o blancos a seguir se determina bajo alguna regla o conocimiento previo de los objetos que realmente interesan para así, seguir y mantener su trayectoria a lo largo de una secuencia de imágenes. Por ejemplo, en un aeropuerto los objetos de interés a observar pueden ser personas únicamente y no algunos carros como los utilizados por los maleteros o los contenedores de basura que maneja el personal de limpieza. La discriminación entre objetos móviles detectados en un escenario corresponde a un nivel posterior a la detección y que se conoce como de asociación de datos.

3.2.1. Método de diferencias de imágenes

La detección de movimiento con fondo estático puede realizarse de diferentes maneras. Algunas son más simples que otras y su uso depende del tipo de información que devuelve la detección. El algoritmo más simple que se puede aplicar, tomando como ventaja el fondo estático o la cámara fija, es el algoritmo de diferencia de imágenes.

De manera formal, sea f_1 y f_2 dos imágenes consecutivas en la línea del tiempo y obtenidas por la cámara que observa el escenario. La detección de movimiento se puede definir como:

$$M(i, j) = \begin{cases} 1 : f_1(i, j) - f_2(i, j) \geq Umbral \\ 0 : f_1(i, j) - f_2(i, j) < Umbral \end{cases} \quad (3.2.1)$$

Donde la matriz binaria M será utilizada en una siguiente etapa por un algoritmo de conectividad de regiones. De hecho, la matriz binaria M , sin importar el algoritmo de detección que se utilice, debe ser utilizada en una siguiente etapa de identificación de regiones o *algoritmo de conectividad* para identificar zonas donde hubo movimiento. La figura 3.1 muestra el resultado obtenido en M como una imagen binaria tras aplicar la ecuación 3.2.1.

Un problema con este algoritmo sencillo es que el nivel de la detección depende mucho del umbral, el cual debe ser establecido empíricamente de acuerdo al esce-



Figura 3.1: Diferencia de imágenes para detectar movimiento, en orden: imagen anterior, imagen actual, diferencia de actual con anterior, y diferencia absoluta

nario a analizar. Otro inconveniente es que es difícil determinar el tamaño del objeto en movimiento. En síntesis, este algoritmo es ineficiente para detectar movimientos pequeños y para estimar el tamaño del objeto que se mueve lo cual puede ser importante para la etapa de asociación de datos.

Una mejora muy efectiva de este algoritmo puede ser el utilizar la diferencia absoluta como lo muestra la siguiente ecuación:

$$M(i, j) = \begin{cases} 1 : |f_1(i, j) - f_2(i, j)| \geq Umbral \\ 0 : |f_1(i, j) - f_2(i, j)| < Umbral \end{cases} \quad (3.2.2)$$

En la figura 3.1, en la última imagen de izquierda a derecha, se muestra el resultado de la matriz de detección de movimiento M . En esta imagen se puede apreciar que el aplicar el valor absoluto a la diferencia permite conocer las diferencias entre un cuadro y otro y no sólo las diferencias en el cuadro actual o anterior. Esto permite obtener una mejor aproximación del área del objeto que se desplaza en la imagen.

A pesar de que esta solución puede resultar económica en costos computacionales y efectiva para la detección aún bajo cambios de luz o cambios graduales en el fondo, aún sigue siendo una solución deficiente para movimientos pequeños y la estimación del área del objeto de interés puede ser un poco excedida ya que en la caja envolvente o el rectángulo que señale al objeto en movimiento, puede incluir áreas del objeto que no existen ya en el cuadro actual debido a que fueron detectados como diferentes con respecto al frame anterior.

3.2.2. Extracción y modelado del fondo

Existen soluciones más sofisticadas en la bibliografía [?] como el cálculo del *flujo óptico* el cuál modela el movimiento de los objetos en la escena y la idea inversa que consiste en modelar el fondo del escenario que permanece estático para detectar posteriormente, a los objetos que se desplacen.

Existen diversos métodos para la extracción del fondo como *la segmentación basada en diferencias acumulativas* [?], *el modelado probabilístico con mezclas gaussianas* [?] o bien *la umbralización mediante el uso del promedio y la desviación estándar*.

Todos estos métodos se basan en observar por un tiempo determinado el escenario de interés con la idea de generar un modelo de cada pixel. Esto es, para cada pixel $f(i, j)$ se genera una función que determina o modela el comportamiento de él a lo largo de observar t imágenes capturadas por la cámara.

La umbralización mediante el uso del promedio y la desviación es un algoritmo muy simple y a diferencia del método de diferencias acumulativas, el promedio y la varianza no requieren almacenar N imágenes. Por otra parte, el método de mezclas gaussianas ha resultado efectivo [?] pero su cálculo puede resultar costoso dependiendo del número de gaussianas que se elijan para modelar cada pixel.

Para obtener el promedio por pixel, la siguiente ecuación puede ser calculada para cada pixel que se observa en las N imágenes que se determinan para obtener el modelo del fondo. En esta ecuación f_k representa la k -ésima imagen observada por la cámara y μ representa la matriz de promedios por cada pixel en las coordenadas (i, j) de la imagen.

$$\mu(i, j) = \frac{1}{N} \sum_{k=1}^N f_k(i, j) \quad (3.2.3)$$

Para la desviación estándar, el desarrollo de la fórmula convencional evita almacenar las N imágenes observadas necesarias para obtener el modelo.

$$\sigma(i, j)^2 = \frac{1}{N} (f_k(i, j) - \mu(i, j))^2 \quad (3.2.4)$$

$$= \frac{1}{N} \sum_{k=1}^N (f_k(i, j)^2 - 2f_k(i, j)\mu(i, j) + \mu(i, j)^2) \quad (3.2.5)$$

$$= \frac{1}{N} \sum_{k=1}^N f_k(i, j)^2 - 2\frac{1}{N} \sum_{k=1}^N f_k(i, j)\mu(i, j) + \frac{1}{N} \sum_{k=1}^N \mu(i, j)^2 \quad (3.2.6)$$

$$= \frac{1}{N} \sum_{k=1}^N f_k(i, j)^2 - 2\mu(i, j)\mu(i, j) + \frac{1}{N} \sum_{k=1}^N \mu(i, j)^2 \quad (3.2.7)$$

$$= \frac{1}{N} \sum_{k=1}^N f_k(i, j)^2 - 2\mu(i, j)^2 + \frac{1}{N} N\mu(i, j)^2 \quad (3.2.8)$$

$$= \frac{1}{N} \sum_{k=1}^N f_k(i, j)^2 - 2\mu(i, j)^2 + \mu(i, j)^2 \quad (3.2.9)$$

$$= \frac{1}{N} \sum_{k=1}^N f_k(i, j)^2 - \mu(i, j)^2 \quad (3.2.10)$$

$$\sigma(i, j) = \sqrt{\frac{1}{N} \sum_{k=1}^N f_k(i, j)^2 - \mu(i, j)^2} \quad (3.2.11)$$

De la ecuación anterior, se puede observar la oportunidad para implementar un algoritmo iterativo para generar el modelo de fondo para cada pixel, únicamente almacenando en acumuladores la información de cada pixel. Un acumulador para la sumatoria de la información de cada pixel y otro para la sumatoria de la información elevada al cuadrado. En la n -ésima iteración, se divide por N y por un lado se obtiene el promedio, por el otro se obtiene el promedio de los cuadrados al que se le resta el promedio elevado al cuadrado y se saca la raíz, esto dará la desviación. Así, la siguiente ecuación se aplicará a una imagen f para construir la matriz de movimiento en dicha imagen, de acuerdo a este modelo:

$$M(i, j) = \begin{cases} 1 : |\mu(i, j) - f(i, j)| \geq \alpha\sigma(i, j) \\ 0 : |\mu(i, j) - f(i, j)| < \alpha\sigma(i, j) \end{cases} \quad (3.2.12)$$

El factor α puede ser determinado de acuerdo a la teoría estadística la cual es-

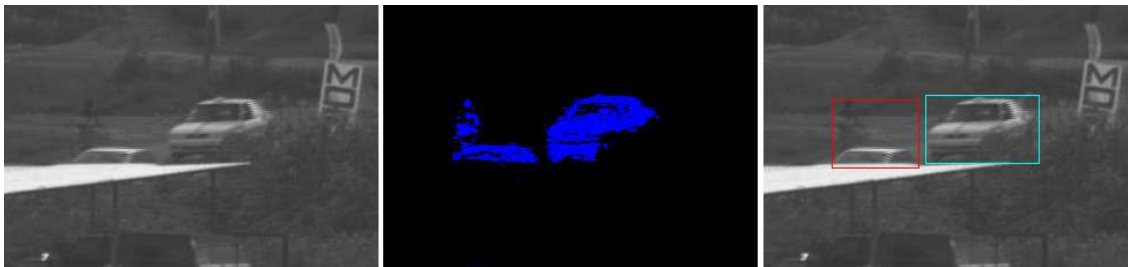


Figura 3.2: Detección de objetos en movimiento utilizando el promedio y la desviación estándar de las observaciones realizadas sobre los píxeles. La imagen de la izquierda corresponde a la imagen original, la del centro a la detección realizada por el modelo y la derecha a la detección que se le muestra al usuario en la interfaz.

tablece que un porcentaje del 99% de los datos observados cae dentro de un intervalo de más o menos α desviaciones estándar alrededor de la media, si ésta toma el valor de 3, asumiendo una distribución normal.

Este método para la extracción de fondo fijo es iterativo, simple y bajo en costo de procesamiento. Su desempeño es mejor comparado a la diferencia de imágenes y además permite una actualización cada N imágenes ya que, al mismo tiempo de observar los píxeles en estas imágenes, se puede ir almacenando la información en los acumuladores correspondientes para que, llegada la n -ésima imagen, se actualice la media y la desviación de acuerdo a las ecuaciones 3.2.3 y 3.2.11.

El modelo que se genera bajo el método descrito tiene ciertas debilidades por ejemplo, el factor α debe ser definido empíricamente de acuerdo al escenario y tipo de objetos a detectar. La detección es sensible al número de imágenes que se observan y en consecuencia, la actualización puede tardar y no reflejarse de manera inmediata en el modelo el comportamiento real del escenario, sobre todo en exteriores en donde puede surgir un cambio de iluminación brusco o repentino. Una ventaja de este método, al igual que cualquiera que modele el fondo, es que en su mayoría permite extraer o estimar la forma del blanco que se mueve, aunque para este caso, el tiempo de procesamiento puede ser menor que el requerido para mezclas gaussianas y más robusto que la simple diferencia o las diferencias acumulativas.

Para la detección, este algoritmo puede ser complementado con los siguientes algoritmos [?]:

- *Algoritmos de Erosión y dilatación.* Utilizados para eliminación de ruido o falsos

blancos de movimiento.

- *Algoritmo de componentes conectadas.* Utilizado para identificar zonas de pixeles conexos discriminados por 1's en la matriz M . Con esto se determina la caja mínima envolvente.
- *Umbralización por área.* Una vez que se han identificado blancos en movimiento, se puede aplicar una discriminación por área ya sea, utilizando el área obtenida por las componentes conectadas o el área de la caja mínima envolvente.

La figura 3.2 muestra un ejemplo de objetos en movimiento detectados mediante el modelo antes descrito.

3.3. Detección del objeto de interés

En el seguimiento de objetos, una etapa importante es la detección o reconocimiento del objeto de interés en el escenario que se observa. Para ello, se debe contar con un modelo predefinido del objeto de interés o bien, un conocimiento previo que lo caracterice de tal forma que pueda ser localizado en una imagen observada por la cámara y asumiendo que el objeto se encuentra presente en la escena.

De manera general, existen dos categorías de métodos utilizados para este fin:

- *Métodos basados en características*
- *Métodos basados en apariencia*

3.3.1. Métodos basados en características

Esta categoría propone construir un modelo del objeto de interés basado en un conjunto de características estructurales que pueden derivarse de la forma misma del objeto. Estas características estructurales pueden ser extraídas de forma manual por el observador o bien, con la ayuda de algún algoritmo que las reconozca y enumere de manera automatizada.

Convencionalmente, las siguientes características pueden ser utilizadas para construir el modelo ya sea en conjunto, en combinación o de manera individual:

- Esquinas.
- Bordes o contornos.
- Puntos de máxima curvatura.
- Intersecciones.
- Centroide.
- Tamaño del área bidimensional.
- Aproximación circular (radio del área).
- Estadísticas extraídas de estas u otro tipo de características.

Estas características tienen por objetivo compactar y reducir la cantidad de información necesaria para representar al objeto de interés. Por otra parte, pueden llegar a ofrecer una representación más robusta ante cambios de iluminación, información redundante o incluso cambios de tamaño del objeto comúnmente derivados ya sea por el acercamiento o alejamiento a la cámara [?].

Una de las debilidades de éstos métodos es la selección y extracción adecuada de las características las cuales, tienen que ser suficientes para distinguir al objeto de interés de otros objetos en la escena. Esto supone, como ya se dijo, un conocimiento a priori del objeto de interés que además debe ser considerado en diversas vistas. Esto es, características que puedan describir al objeto y que además puedan ser actualizadas a los cambios de forma o de posición del objeto. De otro modo, el tomar en cuenta un sólo criterio de características o uno que no represente lo suficiente al objeto, podría llevar a confundirlo con otros objetos similares en la escena. Por ejemplo, un sistema de vigilancia de automóviles colocado en un estacionamiento, podría confundirse al ubicar un automóvil específico si sólo se toman en cuenta las esquinas que lo conforman. ¿ Cuántos automóviles poseen un número de esquinas similares?

Otro punto crítico puede ser el método o los métodos que se elijan para obtener de manera automatizada las características de los objetos para posteriormente, llevar a cabo una clasificación y búsqueda. De hecho, para estos métodos, es de vital importancia la etapa inicial en el sistema que indique el área o ventana de trabajo que señale

la región donde se puede encontrar el objeto de interés. Por ejemplo, un algoritmo de detección de blancos basado en detección de movimiento como los algoritmos explicados en la sección anterior. En suma, del algoritmo de detección de blancos puede depender la sensibilidad o éxito de los métodos basados en características para llevar a cabo la detección.

3.3.2. Métodos basados en área

En estos métodos, la información a nivel pixel y la información cromática que conforman al objeto de interés en la imagen digital, son tomadas como el modelo mismo del objeto. En el caso de los métodos anteriores la información a nivel pixel es tomada en cuenta para obtener el modelo, pero la diferencia es que dicha información es utilizada para crear una estructura de mayor nivel de abstracción con una eliminación de información y creación de estructuras para representar al objeto. En este caso, los métodos basados en área utilizan la información contenida en los pixeles del objeto sin suprimirla. Cualquier comparación con algún blanco, identificado en la escena, implica una comparación o revisión tomando en cuenta punto a punto del modelo con la información a nivel pixel del blanco.

Los métodos basados en apariencia presentan fuertes debilidades ante los cambios de tamaño, de rotación y cambios bruscos del escenario como en la iluminación, ambientes confusos o bien, con la presencia de continuas oclusiones. Las oclusiones de hecho, son un problema que afecta a ambos métodos de detección, basados en características o basados en área. Sin embargo, al ser el modelo conformado por la información del pixel, el éxito de la detección o distinción del objeto, de otro objeto que ocluya o se presente como un posible blanco, depende mucho de los algoritmos de comparación y búsqueda del modelo en la escena.

Pese a ello, los métodos basados en apariencia son muy utilizados para diversas tareas de reconocimiento y seguimiento y se han desarrollado diversos algoritmos que explotan al máximo la información obtenida por los pixeles para realizar la detección.

Una ventaja de estos algoritmos radica en la sencillez del modelo construido con la información de los pixeles y en que la estrategia de comparación del modelo contra un blanco candidato puede ser muy simple. Otra ventaja y por la que son también muy utilizados, es que permiten reconocer y detectar objetos no rígidos como personas

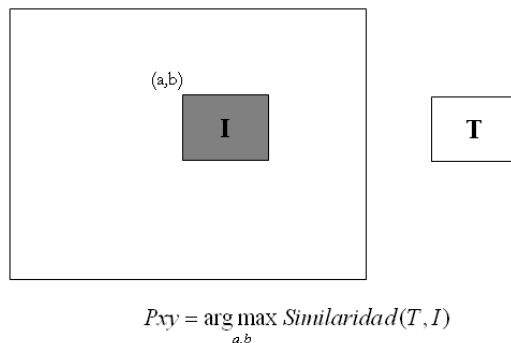


Figura 3.3: Detección de un objeto utilizando un modelo basado en el área

o animales. Del mismo modo, con cambios graduales de la forma del objeto lo cual se puede conseguirse al ir actualizando el modelo bajo alguna estrategia que indique cada cuanto tiempo ésta actualización debe realizarse y si debe ser parcial o total. Además, esta actualización no es costosa ya que sólo se debe llenar nuevamente la matriz que modela al objeto.

Dependiendo de las condiciones del escenario y el tipo de objetos, el algoritmo de búsqueda y comparación puede ser muy sencillo o bien, dependiendo del objeto a seguir y el ambiente del escenario, puede ser muy elaborado. Un ejemplo de esto puede ser utilizar como modelo el histograma cromático del objeto de interés (en todos los canales de color o sólo en la intensidad). De este modo, el algoritmo de comparación consistirá en extraer el histograma cromático del blanco candidato y comparar ambos para medir la similitud de uno con otro. Claramente este método puede resultar muy débil considerando que pueden existir varios objetos con distribución cromática similar.

Medidas de similitud

Precisamente, como en el ejemplo anterior, el algoritmo de búsqueda debe ser complementado con alguno de comparación o de medida de *similitud* que indique de manera cuantitativa la similaridad de un blanco candidato detectado con respecto del modelo. Para ello se han propuesto diferentes medidas de similitud las cuales utilizan como modelo una matriz o subimagen de la imagen obtenida por la cámara. En dicha matriz se almacena la información por pixel que conforman al objeto de interés. Esta

información corresponde a la intensidad cromática del pixel.

Para llevar a cabo la búsqueda en la imagen, se eligen posiciones sobre la misma donde se considere que existe un posible blanco candidato. Para cada posición o coordenada (a, b) candidata, se extraerá una subimagen I o matriz de datos que representarán a un blanco candidato. Por otra parte, T representará a la subimagen o matriz modelo del objeto de interés tomado a priori y T e I cuentan con las mismas dimensiones. La figura 3.3 muestra un ejemplo de un modelo T y una subimagen de las mismas dimensiones que T . A ambas subimágenes o bien, matrices, se les aplica la función de similaridad o *correlación* entre imágenes, para determinan el nivel de semejanza. La posición del objeto quedará determinada por la posición (a, b) en donde la medida de similitud haya devuelto el valor más alto.

En la literatura se pueden encontrar diversas medidas de similitud como la medida *Sum of Absolute Differences SAD*, que es la más sencilla y de bajo costo computacional. La ecuación 3.3.1 describe como se aplica la función de similaridad a dos matrices o funciones discretas en acorde con lo explicado en los párrafos anteriores.

$$C(a, b) = \sum_x \sum_y |T(x, y) - I(x + a, y + b)| \quad (3.3.1)$$

El desempeño de esta medida puede verse disminuido cuando se presentan cambios fuertes de iluminación o bajo contraste. Se pueden enunciar medidas más resistente como la *Sum of Square Absolute Differences* la cual eleva al cuadrado la diferencia de la ecuación 3.3.1 antes de sumarla. También se puede mencionar a la *correlación cruzada normalizada* o la *correlación basada en el error cuadrático medio*.

En este trabajo se optó por utilizar la *correlación con media substraída CMS* la cual ha sido utilizada en diversos trabajos con buenos resultados [?]. Todas las correlaciones o funciones de similaridad se ven afectadas por los cambios de luz o bajo contraste pero la *CMS* se ha mostrado como una medida más robusta que, en combinación con otras técnicas, puede resultar de mayor efectividad que las anteriores [?]. La siguiente ecuación corresponde a la *CMS* cuyo rango va de $[-1, 1]$, con -1 como la similaridad más baja y 1 como la más alta.

$$C(a, b) = \frac{\sum_x \sum_y [T(x, y) - \bar{T}][I(x + a, y + b) - \bar{I}]}{\left\{ \sum_x \sum_y [T(x, y) - \bar{T}]^2 \sum_x \sum_y [I(x + a, y + b) - \bar{I}]^2 \right\}^{\frac{1}{2}}} \quad (3.3.2)$$

En la ecuación 3.3.2 \bar{T} y \bar{I} representan el promedio o media cromática de la subimagen modelo y la subimagen del blanco candidato. Como puede apreciarse también de la misma ecuación, el tiempo de procesamiento es mayor con respecto a otras medidas de similitud. Sin embargo, por su desempeño es preferible sacrificar tiempo de procesamiento por precisión en la detección.

3.4. Ventanas de búsqueda

El cómo se elige o extrae el modelo de interés depende de la aplicación. Por ejemplo, el modelo podría ser extraído mediante una interfaz de usuario la cual, con ayuda del puntero de un indicador (mouse, joystick o algún otro apuntador) seleccione una ventana o región sobre la pantalla en donde se observa el video obtenido por la cámara. Esta selección corresponderá al objeto de interés sobre el cuál se construirá el modelo de acuerdo a alguno de los métodos mencionados en las secciones anteriores. Otro modo sería construir a priori dicho modelo como se ha explicado.

Adicionalmente a la etapa de inicialización del modelo, existe una problemática involucrada por la restricción del tiempo real. La necesidad de obtener una respuesta en la detección bajo una cota de tiempo limitada imposibilita que la búsqueda del objeto de interés pueda realizarse sobre toda la imagen.

Como se ha explicado, en un sistema de vigilancia la etapa de detección de movimiento puede ser utilizada como un primer filtro de eliminación de datos para seleccionar únicamente aquellas regiones o posiciones en donde existe la posibilidad de que el objeto de interés corresponda con alguno de los que presentan movimiento en la escena. Una vez que el objeto de interés ha sido ubicado en la imagen no es necesario volver a utilizar el módulo de detección de movimiento para proponer posibles blancos.

En realidad, aquí hay que hacer la observación de que el objeto de interés posee una dinámica que produce un cambio de posición gradual y continuo, o al menos eso

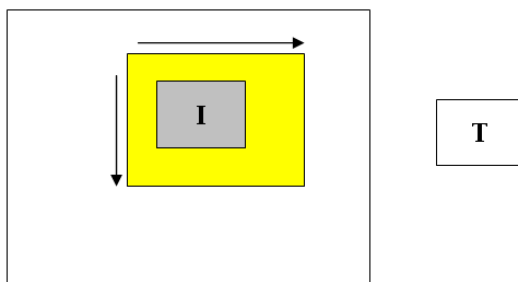


Figura 3.4: Delimitación del área de trabajo mediante el uso de una ventana de búsqueda denotado con el color amarillo

se espera de la mayoría de los cuerpos que se siguen en los sistemas de vigilancia. El cambio de posición continuo en imágenes digitales puede entenderse como una traslación de una posición dada en un instante de tiempo dado. Así, siendo (x, y, t) la posición del objeto de interés en el tiempo t , su cambio de posición en el siguiente instante de tiempo estará dado por $(x + h, y + k, t + l)$. Donde h y k se esperan constantes, de cambio lineal, o de algún otro tipo que pueda ser estimado, es decir que no tengan cambios abruptos o caóticos.

Debido a que los incrementos en h y k no son discontinuos, existe una vecindad de puntos alrededor de la posición (x, y, t) en donde el objeto de interés podría posicionarse en el siguiente instante de tiempo. Esto lleva a la conclusión de que no es necesario buscar sobre toda la imagen, sino sobre una vecindad o ventana de trabajo. A esta estrategia se le conoce como *ventana de búsqueda* y es utilizada en la mayoría de los sistemas de seguimiento. Su uso reduce considerablemente el espacio de trabajo y por ende el tiempo de procesamiento. De hecho, utilizar ventanas de búsqueda resulta adecuado para los algoritmos basados en apariencias. La ventana de búsqueda conduce a una fácil implementación de un algoritmo de corrimiento para seleccionar la posición a partir de la cual se extraerá la subimagen a comparar con el modelo, ver figura 3.4.

Desafortunadamente, existen algunos inconvenientes al utilizar ventanas de búsqueda. El primero de ellos es el establecimiento del tamaño o radio de esta ventana de búsqueda. En algunos trabajos este tamaño se establece en acorde con la dinámica del objeto y esto es, la proporción de cambio de h y k multiplicado por un factor. En otros trabajos como en [?], se propone tomar el doble del radio de la subimagen

modelo con resultados eficientes.

Por otra parte, aún cuando se establezca de manera adecuada el tamaño de la ventana de búsqueda, existe el problema de la oclusión. Si la oclusión es total y ocurre que es del mismo tamaño que la ventana de búsqueda, entonces la detección dará un falso positivo el cuál permanecerá ciclado a menos que se establezca una estrategia de ampliación del tamaño de la ventana o se cuente con un mantenimiento de trayectoria o un estimador-predicor de trayectoria que indique que el seguimiento enfrenta una oclusión. La inclusión de alguna de los componentes mencionados anteriormente exigirán mayor tiempo de procesamiento lo que puede no ser una opción para una aplicación con recursos limitados.

3.5. Estimación de la trayectoria

Existen diversas complicaciones al aplicar la detección sobre una ventana de trabajo mediante algún algoritmo basado en características o basado en apariencia. Algunas de ellas son los cambios de iluminación, el bajo contraste, el cambio de forma y rotación del objeto, y las oclusiones.

Las oclusiones se presentan como el problema más arduo de enfrentar para llevar a cabo la detección. Un buen algoritmo de detección puede enfrentar en gran medida varios de los cambios de apariencias, pero en el caso de la oclusión, la similaridad puede ser baja, para los métodos basados en apariencia, o coincidencias estructurales bajas, si la detección se hizo por características. Esta situación lleva a la pregunta de qué hacer al obtener valores de detección bajos y sobre todo durante un intervalo de tiempo o bien, durante una secuencia de imágenes. Cómo diferenciar que el objeto ha desaparecido y cómo tener la certeza de que no aparecerá en alguna otra posición en la escena. Incluso, si se tuviera idea de que reaparecerá, en dónde buscar. Para este tipo de situaciones, el uso de una ventana de búsqueda se complica también ya que la ventana tiene que ser ampliada para obtener un mejor campo de búsqueda. Esto puede reducir el rendimiento del seguimiento en tiempo real.

Un ejemplo de este problema es el siguiente: imagínese un escenario en donde un objeto se detecta en cada imagen de acuerdo a un modelo construido a priori. La detección se realiza a partir de una búsqueda sobre una región determinada. Ahora

bien, qué ocurriría si el objeto pasa por detrás de una casa o algún otro obstáculo que lo ocluye y además, dicho obstáculo tiene dimensiones mayores al área total de la ventana de búsqueda. Como se mencionó, una posible solución es aumentar el tamaño de la región de búsqueda lo cual puede ralentizar el procesos de detección.

Una segunda alternativa es analizar la trayectoria que el objeto ha venido presentando a lo largo de las imágenes analizadas hasta el momento. Para ello, la idea central consiste en tomar en cuenta el comportamiento de la dinámica del objeto y con esto obtener una estimación de comportamiento del mismo en un instante de tiempo posterior.

El filtro Kalman es una herramienta para la estimación y filtrado de procesos estocásticos lineales. El objetivo principal consiste en obtener una estimación del estado de un sistema basado en las mediciones obtenidas del mismo, tal que esta estimación debe ser óptima, sin sesgo y con varianza mínima [?].

Para el caso del seguimiento de objetos, el filtro Kalman ha resultado muy efectivo para realizar una estimación de la posición del objeto de interés donde la detección del objeto de interés es utilizada como la medición del sistema.

De manera simplificada, el filtro Kalman es un algoritmo recursivo que toma las mediciones de un instante de tiempo $k - 1$ para estimar las mediciones del instante k . Se asume que el intervalo de muestreo es el mínimo necesario para considerar que, el movimiento de características entre dos imágenes consecutivas de la secuencias es lineal. La formalización del filtro se da de la siguiente forma [?]:

Sea el punto $\mathbf{p}_k = [x_k, y_k]$ la posición del objeto detectada en el instante t_k , moviéndose a la velocidad $\mathbf{v}_k = [v_{x,k}, y_{y,k}]$. El movimiento del objeto en la imagen puede entonces describirse con el siguiente vector de estado: $\mathbf{x}_k = [x_k, y_k, v_{x,k}, y_{y,k}]$. Considerando un intervalo de muestreo de imágenes lo suficientemente pequeño, el modelo de movimiento del objeto se puede describir como sigue:

$$\mathbf{p}_k = \mathbf{p}_{k-1} + \mathbf{v}_{k-1} + \xi_{k-1} \quad (3.5.1)$$

$$\mathbf{v}_k = \mathbf{v}_{k-1} + \eta_{k-1} \quad (3.5.2)$$

donde ξ_{k-1} y η_{k-1} son ruido blanco Gaussiano con matrices de covarianza \mathbf{Q}_k y \mathbf{R}_k respectivamente. En términos del vector de estado \mathbf{x}_k descrito anteriormente, las

ecuaciones 3.5.2 y 3.5.1 se pueden escribir como sigue:

$$\mathbf{x}_k = \phi_{k-1} \mathbf{x}_{k-1} + \mathbf{w}_{k-1} \quad (3.5.3)$$

con la matriz de estados ϕ y el vector de ruido blanco Gaussiano \mathbf{w} definidos como sigue:

$$\phi_{k-1} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5.4)$$

$$\mathbf{w}_{k-1} = \begin{bmatrix} \xi_{k-1} \\ \eta_{k-1} \end{bmatrix} \quad (3.5.5)$$

El filtro Kalman utiliza una medida basado en un extractor capaz de medir la posición del punto \mathbf{p}_k en cada imagen de la secuencia. De este modo, el modelo de medida del filtro Kalman es el siguiente:

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mu_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p}_k \\ \mathbf{v}_k \end{bmatrix} + \mu_k \quad (3.5.6)$$

donde H_k es la matriz de medida y μ_k es la incertidumbre de la medida modelada como el ruido blanco Gaussiano.

El algoritmo del filtro Kalman está caracterizado por la matrices de covarianza de estado \mathbf{P}_k , \mathbf{P}'_k y la matriz de ganancia \mathbf{K}_k . \mathbf{P}'_k es la matriz de covarianza de la k-ésima estimación del estado $\hat{x}'_k = \phi_{k-1} \hat{x}'_{k-1}$ predicho por el filtro inmediatamente antes de obtener la medida \mathbf{z}_k . \mathbf{P}_k es la matriz de covarianza del k-ésimo estado, \hat{x}_k obtenido por el filtro después de integrar la medida \mathbf{z}_k , con la predicción \hat{x}'_k . Las matrices de covarianza son un modelo cuantitativo de la incertidumbre de \hat{x}'_k y \hat{x}_k . Finalmente \mathbf{K}_k establece la importancia relativa de la predicción \hat{x}'_k y la estimación del estado \hat{x}_k .

Finalmente, se establecen algunas condiciones antes de su aplicación. En el instante k , las entradas están formadas por las matrices de covarianza del ruido del sistema y de la medida en el instante $k - 1$, \mathbf{Q}_{k-1} y \mathbf{R}_{k-1} respectivamente, la matriz de estado

ϕ , la matriz de medida invariante en el tiempo \mathbf{H} y la medida de la posición en el instante k , \mathbf{z}_k . Las entradas de \mathbf{P}_0 se ponen a valores altos de forma arbitraria, ya que se trata de la incertidumbre inicial del sistema.

De este modo, las siguientes ecuaciones son aplicadas para comenzar a estimar el estado del sistema:

$$\mathbf{P}'_k = \phi_{k-1} \mathbf{P}_{k-1} \phi_{k-1}^t + \mathbf{Q}_{k-1} \quad (3.5.7)$$

$$\mathbf{K}_k = \mathbf{P}'_k \mathbf{H}_k^t (\mathbf{H}_k \mathbf{P}'_k \mathbf{H}_k^t + \mathbf{R}_k)^{-1} \quad (3.5.8)$$

$$\hat{\mathbf{x}}_k = \phi_{k-1} \hat{\mathbf{x}}_{k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \phi_{k-1} \hat{\mathbf{x}}_{k-1}) \quad (3.5.9)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k) \mathbf{P}'_k (\mathbf{I} - \mathbf{K}_k)^t + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^t \quad (3.5.10)$$

La salida es la estimación óptima de la posición y la velocidad en el tiempo k , $\hat{\mathbf{x}}_k$, y sus incertidumbres, dadas por los elementos diagonales de \mathbf{P}_k .

Un inconveniente de esta herramienta es que la estimación y filtrado depende mucho del modelo dinámico del objeto de interés, el cual debe ser proporcionado. En el caso del filtro Kalman lineal, los modelos dinámicos corresponden a modelos de movimiento lineales por lo que la estimación puede resultar ineficiente para objetos con movimientos no lineales. Existen variantes del filtro Kalman para hacer posible la estimación en modelos no lineales como el filtro Kalman extendido (EKF) o el Filtro Unscented Kalman (UKF). Para objetos con dinámicas más complejas, se han elaborado herramientas que proponen el uso de diversos modelos dinámicos como lo son filtros de múltiples Modelos Interactuantes (IMM).

Para propósitos de este trabajo, el filtro Kalman es utilizado en alguno de los algoritmos de seguimiento utilizados. Así, se busca ejemplificar el uso de herramientas de estimación en el seguimiento utilizando como espacio de trabajo las imágenes foveales generadas con el método propuesto.

3.6. Algoritmos para seguimiento

Tradicionalmente los sistemas de visión que realizan seguimiento deben trabajar con imágenes de resolución invariante. Como se ha visto, las restricciones de tiempo real imposibilitan un procesamiento completo de la imagen. Bajo estas condiciones, un algoritmo muy simple que se puede utilizar para el seguimiento se ilustra en la figura 3.5.

3.6.1. Algoritmo de seguimiento básico: detección con medida de similitud

De manera breve, la idea consiste en analizar una imagen capturada por la cámara, restringir la zona de trabajo a una región o ventana de búsqueda, y sobre esa ventana aplicar algún algoritmo de detección como los que se han mencionado en las secciones anteriores y bajo la premisa de contar con un modelo a priori del objeto a detectar. Si la detección es satisfactoria, entonces la región de búsqueda debe reubicarse sobre el centro del objeto detectado. De otro modo, no habrá cambios. El seguimiento se conformará con reportar las coordenadas (x, y) del objeto detectado en cada imagen obtenida de la cámara o bien de una secuencia de imágenes.

En cuanto la obtención a priori del modelo del objeto de interés, esto dependerá de la aplicación. Por ejemplo, podría tratarse de una aplicación de vigilancia donde el usuario *enganche* o seleccione de algún modo, una región parcial o total. Otro modo podría ser utilizar un algoritmo de detección de movimiento y seleccionar alguno de los blancos detectados bajo algún criterio, como un objeto de interés.

En este trabajo el algoritmo de detección utilizado se basa en medidas de similitud, específicamente en la *correlación con media abstraída*, por ser una de las medidas de detección con mayor efectividad. En este caso, el modelo del objeto de interés consiste en llenar una matriz de datos conteniendo la información a nivel pixel de la región seleccionada por el usuario o por algún algoritmo de detección indicando el blanco de interés.

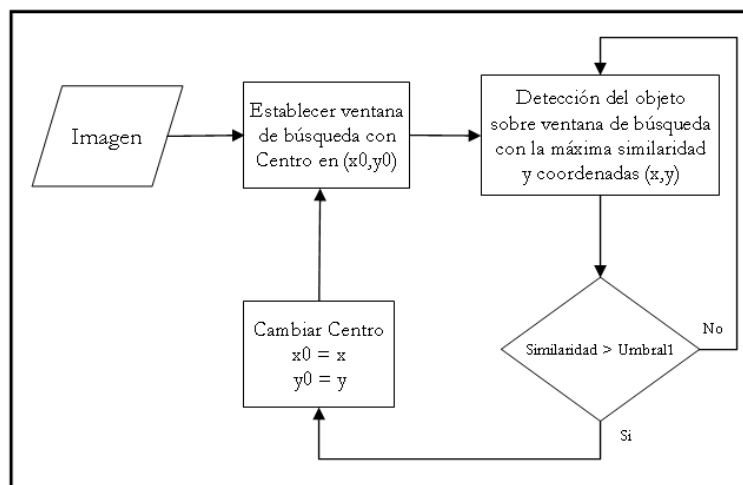


Figura 3.5: Diagrama a bloques de un algoritmo de seguimiento básico

3.6.2. Algoritmo de seguimiento: medida de similitud y detección de movimiento

Como se explicó en la sección anterior, el algoritmo basado en ventanas de búsqueda y medias de similitud cuenta con una debilidad fuerte que es la baja eficacia ante oclusiones. El seguimiento de un objeto se puede considerar exitoso hasta que el objeto mismo es ocluido o este sale de la escena. Pero si el objeto es ocluido parcialmente o bien, la oclusión es momentánea y esto es: el objeto que ocluye se retira o el objeto de interés pasa por detrás de algún obstáculo que lo ocluye, entonces el algoritmo debe detectar nuevamente al objeto de interés y mantener el seguimiento sobre él.

Si el objeto es ocluido parcial o totalmente, esto se verá reflejado en el valor de similitud devuelto por la detección. Si este valor, que se supone es el más alto encontrado, es muy bajo (recordando que la función de similitud va de $[-1,1]$ o bien de $[0,1]$ dependiendo como se programe) entonces puede significar que la posición detectada es un falso positivo y esto es, que la detección es errónea. Para evitar una detección falsa el algoritmo de seguimiento debe contar con una estrategia para reubicar nuevamente al objeto de interés en el escenario.

Una estrategia sencilla a utilizar cuando la similitud es no confiable o baja con respecto a un umbral, consiste en realizar una detección de blancos posibles basado

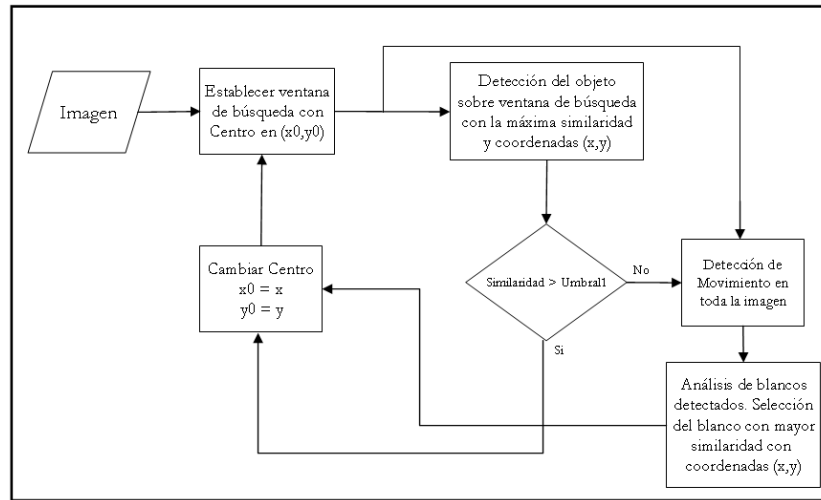


Figura 3.6: Diagrama a bloques del algoritmo de seguimiento en imágenes cartesianas

en detección de movimiento, aprovechando que las condiciones del escenario son de cámara fija o fondo estático. De este modo, cada blanco es analizado por la medida de similitud para identificar al objeto de interés. Este procedimiento es aplicado continuamente hasta que el objeto es localizado nuevamente y entonces se continúa con el seguimiento alrededor de una ventana de búsqueda mientras la similitud se encuentre por arriba del umbral determinado. La figura 3.6 muestra el algoritmo descrito como diagrama de flujo.

El algoritmo anterior es más eficiente que el primero que no enfrenta oclusiones. En realidad, para este algoritmo o cualquier otro que trabaje con ventanas de búsqueda, la oclusión siempre representará un cuello de botella que implicará aumentar el espacio de búsqueda en los datos ofrecidos por la imagen. Esto es, la ventana de búsqueda deberá ser ampliada o bien, como en el algoritmo anterior, alguna estrategia de detección de blancos probables debe ser empleada para ampliar la búsqueda del blanco en la imagen. Para ello hay algunas interrogantes: cuando se usan ventanas de búsqueda y ocurre una oclusión, ¿qué tanto debe ampliarse la ventana de búsqueda?, para un módulo de detección de blancos ¿qué blancos deben ser analizados, es necesario que se analicen todos los detectados, o sólo los que cumplen ciertas características? La respuesta de estas preguntas en el algoritmo de seguimiento deberán ser elegidas con cuidado ya que un procesamiento de datos elaborado puede mellar el desempeño del

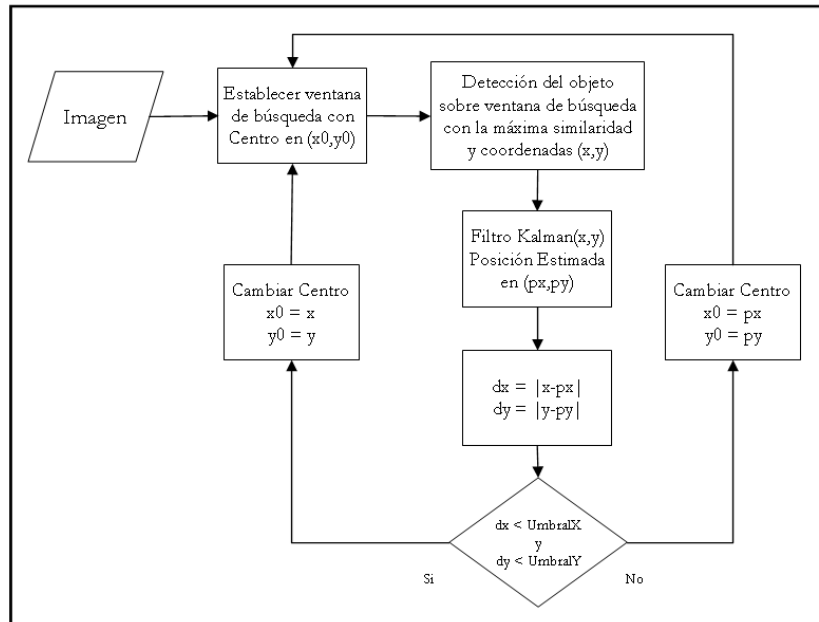


Figura 3.7: Diagrama a bloques del algoritmo de seguimiento en imágenes cartesianas con un filtro Kalman incorporado para estimación de trayectoria

algoritmo que debe trabajar en tiempo real.

3.6.3. Algoritmo de seguimiento con estimación de trayectorias

El filtrado y estimación de trayectorias se presenta como una herramienta que permite mantener la trayectoria del objeto aún cuando se presenten oclusiones. Esto evita el pensar en estrategias de ampliación de la ventana de búsqueda o en estrategias de detección de blancos ya que con la estimación es posible conocer con cierto error la posición probable en donde el objeto puede aparecer. De este modo, la ventana de búsqueda se colocará sobre dicha posición y el algoritmo de seguimiento continuará trabajando normalmente. La figura 3.7 muestra un diagrama de flujo del algoritmo explicado.

3.6.4. Algoritmo de seguimiento para imágenes foveales

Los algoritmos anteriores tienen un buen desempeño para el seguimiento de un objeto y pueden robustecerse con algunas estrategias adicionales dependiendo de la complejidad del escenario y la forma misma del objeto de interés. En el caso de utilizar imágenes foveales, el modo de detección y el espacio de trabajo varían debido al factor de la resolución variable sobre la imagen.

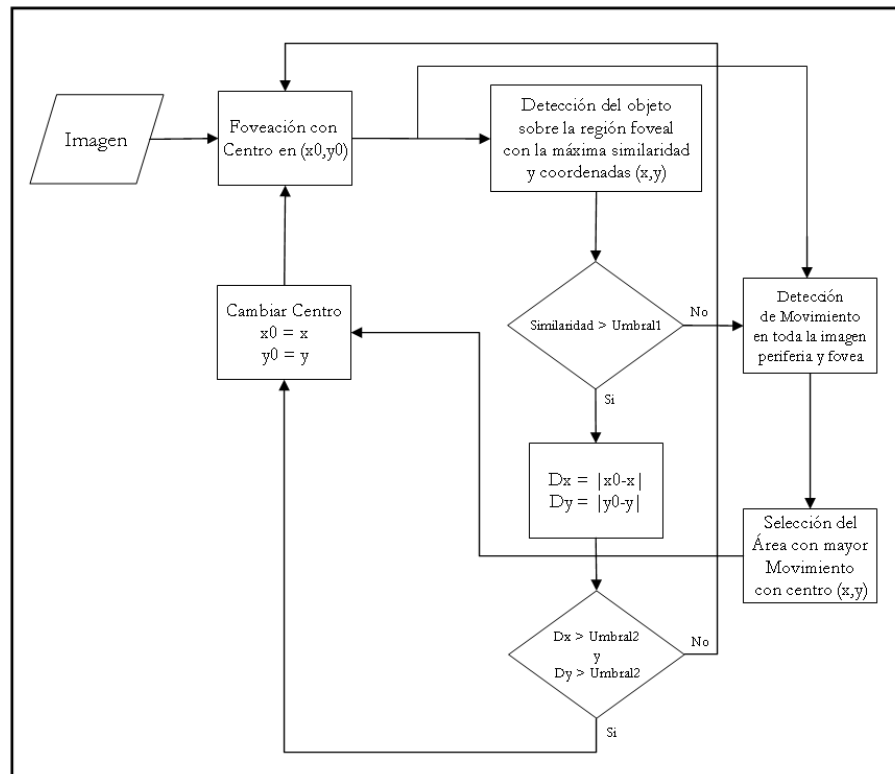


Figura 3.8: Diagrama a bloques del algoritmo de seguimiento incorporando el uso de las imágenes foveales

Como primer ventaja de utilizar imágenes foveales se tiene la reducción de datos a procesar. Además, otra ventaja es que el muestreo variable incorpora un nivel de detección jerárquica con respecto al centro o área de mayor resolución de la imagen. Lo anterior significa que conforme un objeto se aleja del centro hacia la periferia, la información que se tenga de él irá decreciendo en proporción a esta distancia y es probable que, al encontrarse muy a la orilla, su presencia pase desapercibida a causa

del bajo muestreo sobre dicha región. De aquí se desprende la condición de mantener al objeto de interés lo más que se pueda al centro de la imagen o bien, sobre la región de mayor muestreo o región foveal. Así, se contará con información de calidad para su análisis y detección mientras que, para el caso de los objetos a su alrededor, su importancia se verá reflejada en cuanto a su cercanía con el objeto de interés o bien su cercanía a la región foveal.

La reducción de datos y la detección jerárquica son dos puntos claves para realizar el seguimiento en las imágenes foveales. La manera de utilizarlos se inspira también del modo en que el seguimiento es realizado por los sistemas de visión natural. De acuerdo a Schwartz [?], en los sistemas visuales la región foveal se utiliza para analizar al objeto de interés con mayor detalle. La textura, forma e intensidad pueden ser analizadas gracias a la alta resolución con la que esta área cuenta. En la zona periférica, la información visual es procesada con mucho menos detalle, pero en forma sumamente rápida. Ambos tipos de procesamiento funcionan en forma complementaria. De esta forma, cuando un objeto aparece repentinamente en la periferia del campo visual, el sistema visual lo detecta, y lo centra respecto a la fovea (foveación). Una vez centrado, el sistema visual lo procesa en detalle. La detección de los objetos en la periferia del campo visual se realiza combinando mecanismos de procesamiento de bajo y alto nivel los cuales se explican a mayor detalle en [?].

La fijación y los movimientos sacádicos, como se explicaba en la sección 1.4.1, son los movimientos que usa el sistema de visión foveal natural en los primates. Como se explicó ampliamente con anterioridad, la fijación es el análisis meticuloso sobre el objeto de interés (detección) y los movimientos sacádicos se refieren al posicionamiento de la región de atención o foveal sobre algún objeto en el campo de visión detectado en la periferia.

Si se analiza la descripción anterior, esto corresponde a un algoritmo de seguimiento muy similar a los descritos para imágenes con resolución invariante. La clave aquí es mantener al objeto de interés sobre la región foveal, que en las imágenes cartesianas corresponde a centrar la ventana de búsqueda sobre la coordenada donde se detecta al objeto de interés. Por otra parte, la periferia como se ha explicado, se utiliza para detectar posibles blancos de interés basados en algunos procedimientos que permiten un procesamiento veloz que a su vez se origina de procesar menos datos que en el centro. Esta detección puede variar dependiendo de las necesidades de la aplicación.

En este trabajo, esta detección consiste en realizar la detección de movimiento la cuál, por la menor cantidad de datos en la imagen foveal, se procesará con mayor velocidad que en las imágenes cartesianas.

Así, tomando como referencia el diagrama de la figura 3.6, el algoritmo de seguimiento a aplicar sería: enganchar u obtener un modelo a priori de objeto de interés y colocar la región foveal sobre el objeto de interés ubicado en la escena. Detectar al objeto y mantener a la región foveal sobre el objeto de interés. Aquí es importante señalar que la región puede cubrir un área similar a la cubierta por la ventana de búsqueda en las imágenes cartesianas. Por lo tanto, la detección se realiza únicamente sobre la zona foveal.

En el momento que una oclusión se presenta, si la similaridad indica una pérdida del objeto de interés, entonces se puede pasar a la etapa de detectar movimiento para indicar posibles blancos. En este caso, la jerarquía del muestreo se verá reflejada ya que claramente los objetos con mayor movimiento tenderán a encontrarse muy cercanos al centro.

Debido a que se asume que el muestreo foveal se aplica a la salida de una cámara ¹, entonces la región foveal puede moverse una sola vez por cada captura. Una heurística propuesta consiste en mover la zona o región foveal a aquella posición detectada en la región periférica en donde se halló el mayor movimiento. En el siguiente instante de tiempo o siguiente imagen capturada por la cámara, el algoritmo procederá a analizar la imagen en la región foveal mediante la detección. Si el blanco detectado resulta ser el objeto de interés entonces se pasará a la detección en la región foveal. De otro modo, se continuará analizando la región periférica y la foveal bajo el criterio de buscar el mayor movimiento detectado.

Si la región foveal se desplaza con el objeto de interés, entonces las imágenes foveales mostrarán un fondo móvil como si la cámara se estuviera moviendo, en el caso de que la fovea se mueva por software. Si el sistema es activo y la cámara se encuentra en movimiento de cualquier modo el fondo permanecerá dinámico. Esto puede impactar negativamente en los algoritmos de detección de movimiento, como los basados en substracción y modelado del fondo. Sin embargo, el método de diferencia

¹Esto se refiere a que no se cuenta con la imagen cartesiana en memoria, sólo con la foveal obtenida ya sea de aplicar la transformada al frame grabber o de obtenerla directamente de la cámara por algún dispositivo de hardware.

absoluta puede resultar efectivo ya que al analizar dos imágenes consecutivas las diferencias entre ellas disminuyen. Esto dependerá de la velocidad de captura de la cámara o framegrabber. Una forma de solucionar esto es no mover la zona foveal en el instante de captura en el que la detección retornó una baja similaridad. Así, al capturar la siguiente imagen, la diferencia de imágenes podrá ser aplicada ya que el fondo no habrá variado (recordando que las restricciones son cámara fija o fondo estático para este trabajo) entre la imagen del instante anterior y la actual. La figura 3.8 muestra el diagrama de flujo del seguimiento de un objeto como se ha descrito e incorpora el uso de imágenes foveales como espacio de trabajo.

Las ideas explicadas muestran la importancia de las imágenes foveales obtenidas por el método propuesto en el Capítulo 4. Esto se debe a que para la detección de movimiento y para los algoritmos de detección que se han explicado en las secciones anteriores, la linealidad así como la relación de adyacencia espacial de los píxeles son importantes. La figura 3.9 muestra un objeto de interés localizado en diferentes posiciones en el escenario y cómo influye su posición para su detección en la imagen foveal. Se puede apreciar que la información del objeto adquirida en la imagen foveal disminuye o aumenta dependiendo de su cercanía a la región foveal. La mayor información se obtiene cuando el objeto se encuentra sobre la región foveal donde la resolución es máxima o igual a la de la imagen cartesiana. En la primera fila de la imagen se muestran las imágenes cartesianas con resolución invariante. En la segunda fila se muestran las imágenes recuperadas de la imagen foveal, las cuales exhiben la colocación de los anillos cuadrados con muestreo variante en cada anillo. La última fila muestra las imágenes foveales de trabajo.

3.7. Discusión

Finalmente, en resumen el seguimiento sobre las imágenes foveales propuestas consiste en aplicar el método de foveación a las imágenes cartesianas, aplicar el algoritmo de seguimiento descrito anteriormente, y obtener con ayuda de la transformación foveal inversa la coordenada cartesiana correspondiente a la detección del objeto en la coordenada foveal. Gracias a esto, un estimador de trayectorias como el filtro Kalman también puede ser utilizado como complemento en el seguimiento utilizando

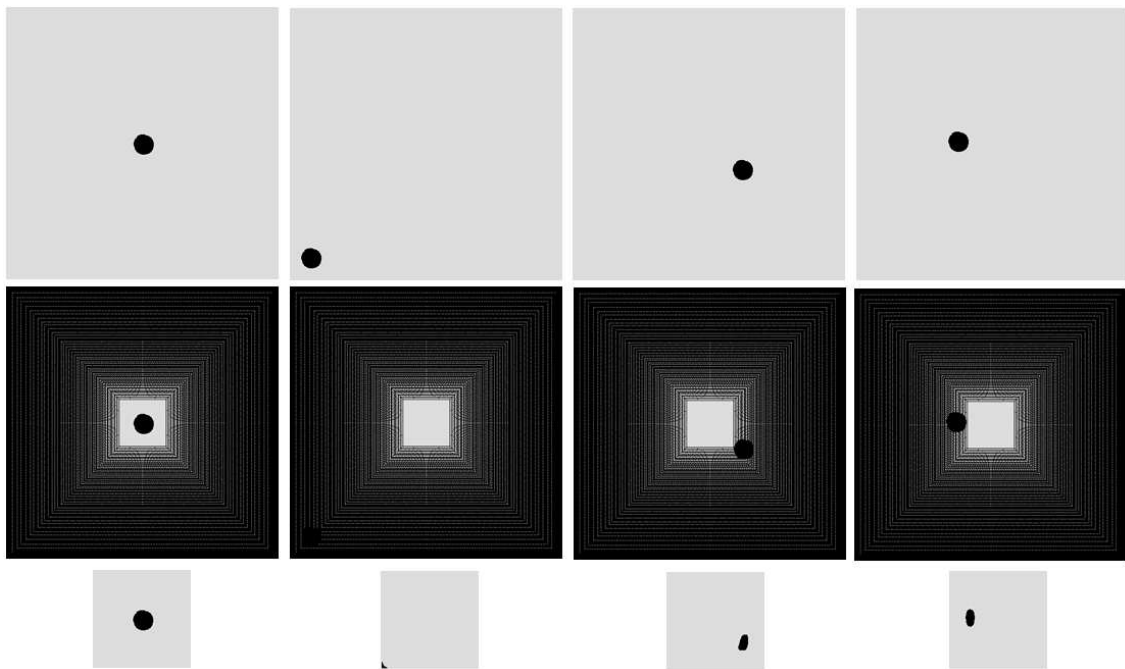


Figura 3.9: Ejemplos que muestran cual es el efecto en la imagen foveal de que un objeto (mancha negra) se encuentre localizado en diferentes regiones sobre el escenario

las imágenes foveales. Para ello, las coordenadas cartesianas obtenidas con la transformación inversa son introducidas al filtro, el cual estimará la trayectoria del centro de la región foveal que se desplaza con el objeto de interés a lo largo de la escena siempre que la cámara permanezca fija. Si se tratase de un sistema activo entonces el filtro Kalman se utilizaría mas bien para estimar el movimiento de la cámara con ayuda de la detección en la región foveal.

En el siguiente capítulo se expondrán y discutirán los experimentos realizados para mostrar la efectividad del seguimiento utilizando las imágenes foveales propuestas y utilizando los algoritmos descritos en este capítulo.

Método de foveación propuesto

En suma, tres son los trabajos más representativos que han propuesto un método de foveación. La Transformada Log Polar, las Geometrías Cartesianas Exponenciales y la Transformada Wavelet Foveal. Las imágenes foveales generadas con cualquiera de estos métodos han sido utilizadas en diversos trabajos en la tarea de seguimiento de objetos. Sin embargo, han surgido algunas limitaciones importantes en cuanto al alcance de las aplicaciones que se han podido desarrollar. Las limitaciones más importantes son [?]:

1. El seguimiento de objetos se reduce al seguimiento de un sólo objeto.
2. En las imágenes foveales generadas por la TLP no es posible aplicar los métodos convencionales de detección y localización de objetos.
3. El método de foveación propuesto por las GCE debe ser implementado en hardware si se desea incorporar a una aplicación con respuesta en tiempo real.
4. La TWF es compleja aún para ser implementada en hardware, lo que impide también su uso en aplicaciones de tiempo real.

En consecuencia, las aplicaciones de seguimiento de objetos trabajan bajo condiciones específicas como fondo fijo o cámara fija, escenarios interiores e incluso, en algunos casos, el objeto de interés debe tener una forma geométrica definida.¹

En este capítulo se presenta una propuesta diferente de foveación para reducir las limitaciones presentadas en los trabajos anteriores, aunque cabe aclarar que al igual

¹Esta restricción surge cuando se realiza el seguimiento en las imágenes Log Polares

que en los otros trabajos, esta propuesta está orientada al seguimiento de un objeto ya que, como se explicó en el la sección 2.2, la visión foveal implica que la atención, detección y seguimiento pueden realizarse solamente sobre un objeto de interés.

La idea está basada en el modelo originalmente propuesto por Tistarelli y Sandini [?], para la implementación del modelo foveal de Schwartz. La idea principal en la propuesta de este trabajo, consiste en utilizar las ecuaciones del modelo de Tistarelli para determinar el muestreo bajo las propiedades foveales de tal modo que, permita preservar la linealidad y a su vez, no implique un proceso complejo que force a su implementación en hardware para ser aplicable en tiempo real.

4.1. Modelo log polar discreto

El modelo log polar discreto [?] propone una discretización de la TLP a partir de su modelo continuo, modificando de manera sencilla las ecuaciones 2.3.1 y 2.3.2. En principio, se asume que, para la distancia radial-logarítmica, se puede utilizar el logaritmo en cualquier base. También toma en cuenta la singularidad de la TLP que se mencionó en el capítulo anterior y que ocurre por la definición propia de los argumentos que recibe el logaritmo. Se asume una distancia mínima ρ_0 y una distancia máxima ρ_{max} a la cual se encuentran posicionados los anillos de los fotorreceptores.

No obstante que en visión por computadora las coordenadas (x, y) , de una imagen digital son discretas, el mapeo realizado por la TLP no produce coordenadas discretas. De acuerdo a lo anterior, la TLP en su modelo continuo toma un punto de coordenadas Cartesianas (x, y) y lo mapea a los intervalos $[\log_a(\rho_0), \log_a(\rho_{max})] \times [0, 2\pi]$. Aún cuando se truncaran los valores obtenidos por la TLP, los valores de los intervalos no permiten un mapeo directo hacia un plano o imagen digital.

En [?] Tistarelli propone las siguientes ecuaciones para la discretización del mapeo de la coordenada discreta (x, y) a la coordenada foveal (ξ, γ) y asumiendo que el centro de la región foveal es colocado sobre la posición (x_0, y_0) .

Primero se realiza la conversión en coordenadas radiales-angulares:

$$\rho = \sqrt{(x - x_0)^2 + (y - y_0)^2} \quad (4.1.1)$$

$$\theta = \tan^{-1} \left(\frac{y - y_0}{x - x_0} \right) \quad (4.1.2)$$

Posteriormente, se calculan las coordenadas foveales discretas (ξ, γ) asumiendo que la malla Log Polar se encuentra dividida en N_r anillos y N_a sectores:

$$\xi = \log_a \left(\frac{\rho}{\rho_0} \right) = \log_a(\rho) - \log_a(\rho_0) \quad (4.1.3)$$

$$\gamma = \frac{N_a}{2\pi} \left(\theta + \frac{\pi}{2} \right) \quad (4.1.4)$$

Donde ξ y γ satisfacen las relaciones $0 \leq \xi \leq N_r$ y $0 \leq \gamma \leq N_a$ para las cuales sólo ciertos valores de (x, y) son válidos. Por otra parte, la base a del logaritmo está en función del número de anillos N_a en los que se desee dividir la malla Log Polar. Su cálculo se realiza como sigue, dado el número de anillos N_r y la ecuación 4.1.3 se tiene que:

$$\begin{aligned} N_r &= \log_a \left(\frac{\rho_{max}}{\rho_0} \right) \\ a^{N_r} &= \frac{\rho_{max}}{\rho_0} \\ \rho_0 a^{N_r} &= \rho_{max} \\ a &= \exp \left(\frac{1}{N_r} \ln \left(\frac{\rho_{max}}{\rho_0} \right) \right) \end{aligned} \quad (4.1.5)$$

Con estas ecuaciones la discretización se puede llevar a cabo de dos modos:

1. Variando a (ξ, γ) en todos sus valores posibles y despejando entonces las ecuaciones 4.1.3 y 4.1.4 para obtener las coordenadas radiales-angulares (ρ, θ) correspondientes. Con estas coordenadas es posible recuperar las coordenadas cartesianas mediante las ecuaciones 2.3.5 y 2.3.6 expuestas en el capítulo 2.
2. Resolviendo las ecuaciones 4.1.3 para cada uno de los valores de las coordenadas

(x, y) . Si la solución arroja valores de (ξ, γ) fuera de los rangos permitidos, entonces (x, y) se descarta. De otro modo, el valor que se obtenga puede ser truncado o redondeado para asignar finalmente el valor de (ξ, γ) . De este modo, varias coordenadas (x, y) coincidirán en una coordenada (ξ, γ) . Para el caso de aplicar este mapeo a una imagen digital, surge la disyuntiva de cual valor de pixel tomar de acuerdo a que coordenada. Una solución inmediata sugiere promediar los valores de pixel de todas aquellas coordenadas que coincidan en el mapeo. Otra puede ser tomar la primer calculada o bien, algún método de interpolación. Esto se queda a la libertad del programador y dependiendo de la precisión de la información en la zona periférica ya que es en esta zona donde se notará el efecto de la estrategia decidida.

4.2. Foveación con anillos cuadrados

Como se ha mencionado, el modelo de Schwartz expresa matemáticamente la distribución de los fotorreceptores en la retina. El problema con este modelo es el mapeo que se realiza cuando el fotorreceptor es identificado por sus coordenadas logarítmicas-radiales y angulares. Como se veía en el capítulo 2, la necesidad de utilizar un método de foveación emergió al querer agrupar de manera ordenada la información obtenida por los fotorreceptores que en este caso, modelan al muestreo que se realiza en la imagen digital. Sin embargo, tanto el modelo de Tistarelli, como cualquier otro basado en la TLP, agrupa la información de tal modo que la relación espacial que se tenía originalmente se pierde.

Por otra parte las GCE brindan una pirámide de imágenes lineales en donde el concepto de región foveal y región periférica queda determinado por el nivel de la pirámide en donde se encuentre procesando. La generación de esta pirámide implica como ya se dijo, tiempo de procesamiento valioso además de algunas complicaciones en la detección y localización del objeto.

En este trabajo se diseñó un método de foveación que incorpore en parte el modelo matemático de Schwartz y a su vez que permita conservar lo mejor posible la relación espacial de adyacencia de los pixeles. Para ello se pensó en utilizar anillos cuadrados como en las GCE, en vez de circulares como en la TLP. Y en vez de pensar

en como muestrear con estos anillos cuadrados para obtener una imagen lineal, se pensó primero en que la imagen lineal ya estaba dada la cual correspondería a la imagen foveal.

Para una mejor explicación, obsérvese la figura 4.1. La imagen de la izquierda representa a la imagen foveal a la que se desea llegar. La cuadrícula del centro representa la región foveal donde se supone habrá una representación con alta resolución de la imagen original. Los alrededores representan a la zona periférica que se supone, se encontrará repartida en proporciones logarítmicas a lo largo de la imagen original. En esta imagen el área foveal es representada por un cuadrado, mientras que los anillos que arrojaron el muestreo se representan por los anillos cuadrados de los cuales algunos se han marcado en tonos rojos. En cada anillo, cada uno de los pixeles o cuadros que lo conforman, representa un fotoreceptor o bien un pixel que tiene el valor de muestreo de la imagen original, ya sea que se haya tomado de manera simple o mediante el promedio de una región en la imagen.

Una vez establecida la estructura de la imagen foveal, se presenta como es que los datos que contienen la información de muestreo son ubicados sobre la imagen original. La idea es simple y consiste en expandir los anillos sobre la imagen original partiendo del centro de la región foveal hacia las orillas de la imagen original. Los pixeles o fotoreceptores contenidos originalmente en cada anillo de la imagen foveal también se expanden sobre el mismo anillo de manera uniforme. Su posición final representa el pixel que va a ser muestreado o el centroide de la región que será muestreada. Esto se ilustra en la figura 4.1 en la imagen de la derecha. El pixel indicado en rojo muestra como se posiciona éste partiendo de la imagen foveal hacia la original.

Hasta este punto, la pregunta que surge es: ¿qué regla seguir o con qué criterio expandir a los anillos cuadrados que conformarán a la imagen foveal?. Aquí es donde se utilizó parte del modelo de Tistarelli para el mapeo discreto de la TLP. La ecuación 4.1.5 indica, para un número de Anillos N_r deseado, cuál es la base del logaritmo a utilizar para determinar los radios de los anillos en la malla Log Polar. Una vez obtenido el valor de esta constante (a), los valores de cada radio se encuentran dados por la siguiente ecuación que se deduce también de los pasos algebraicos para obtener la ecuación 4.1.5:

$$\rho_\xi = \rho_0 a^\xi, \quad \xi = 0 \dots N_r \quad (4.2.1)$$

Esta distancia ρ_ξ es la que se utiliza para definir el radio de los anillos cuadrados. La posición de los píxeles de muestreo se establece en relación de esta distancia ya que se desean colocar de manera uniformemente espaciada sobre el anillo mismo. Esto difiere del modelo discreto de la TLP ya que los fotorreceptores o píxeles de muestreo se colocan alrededor del anillo con una distancia angular uniforme la cual no varía para cada anillo. En la propuesta de este trabajo la división angular si varía en cada anillo, aunque para cada anillo la división entre cada punto de muestreo es uniforme. Esta división dependerá del número de puntos que se van a muestrear y se relaciona con el radio del anillo.

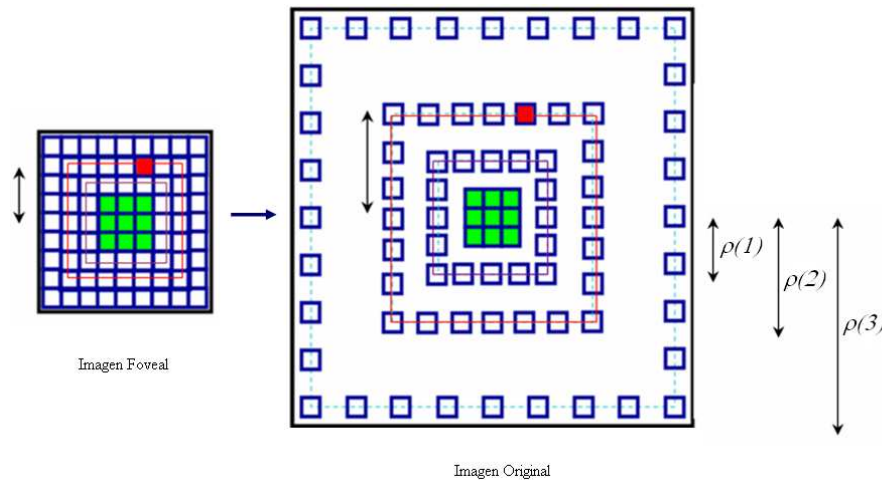


Figura 4.1: Estructura de la imagen foveal a la izquierda y cómo se lleva a cabo el muestreo en la imagen original

Otro punto que se modifica en esta propuesta, con respecto a la del modelo discreto de la TLP, es el factor ρ_0 que indica la distancia al primero anillo y que es utilizada para evitar la singularidad de la TLP. En esta propuesta, esta distancia ρ_0 es utilizada para establecer el radio del área foveal ya que para este método de foveación no existe algún punto de singularidad.

Una vez definida la estructura de la imagen foveal y como se coloca hipotética-

mente sobre la imagen original, se presenta ahora cómo es que un punto en la imagen original debe ser mapeado en la imagen foveal y viceversa.

Observando nuevamente la figura 4.1, en ambas cuadrículas se ha señalado un pixel como un cuadro rojo. Este punto guarda una relación de proporción en ambos anillos al establecer que, tanto en la imagen foveal, como en la imagen original, la separación que hay entre los pixeles o fotoreceptores es uniforme. Por lo tanto, el número de anillo de la imagen foveal indica su distancia con respecto del centro de la imagen al sumarle el radio del área foveal ρ_0 . Por otro lado, en la imagen foveal, la distancia de este anillo esta dada por la ecuación 4.2.1. Así, la siguiente relación se establece:

$$\Delta\rho_\xi = \frac{\rho_\xi}{\rho_0 + \xi} \quad (4.2.2)$$

donde ρ_ξ esta dado por la ecuación 4.2.1.

De este modo, para cualquiera que sea el caso, ya sea que se mapee de la imagen foveal a la original o de la original a la foveal, lo que se requiere saber es el número de anillo al que pertenece la coordenada. De manera directa se conocerá entonces su radio en la imagen original. Con esta información, se podrá obtener el valor correspondiente $\Delta\rho_\xi$ y entonces, se procederá a multiplicar o dividir a las coordenadas por esta proporción.

Es importante señalar que en la práctica la ecuación 4.2.1 indicará que un anillo se coloque a una distancia real sobre la imagen digital, pero al tratarse de una imagen discreta, entonces existe la posibilidad (depende también del radio de la zona foveal) de que los primeros anillos coincidan en valor al ser truncados. Este detalle podría pasarse por alto, pero entonces alrededor de la región foveal se colocarían anillos con información repetida del centro y que podría ser innecesaria, lo cual lleva a indicar que del número de anillos N_r solicitados sólo se colocarán un número RN_r el cual por supuesto es menor.

Una solución sería fijar el radio foveal $p_0 = 1$ y dejar que los anillos definan el radio de la zona foveal. Pero en este trabajo se ha optado por controlar el tamaño de esta región y fijar los anillos en función de este tamaño. Otra posible solución podría ser colocar un número de anillos N_r de tal manera que el factor a permita una

separación discreta ($|\rho_i| \neq |\rho_{i-1}|, i \dots N_r$). Pero este camino no es muy adecuado ya que los anillos podrían quedar muy separados.

Este problema puede resolverse colocando alrededor de la región foveal a aquellos anillos cuya distancia $|\rho_i|$ no se ha repetido con ninguno de los otros anillos anteriormente colocados. Si se hace esto, entonces podría haber un problema al utilizar el factor $\Delta\rho_\xi$ al momento de transformar un punto de las coordenadas foveales a las coordenadas cartesianas correspondientes. La tabla 4.1 muestra un ejemplo del problema que se está discutiendo y cómo la inclusión de una nueva variable puede resolver la asignación de anillos sobre la imagen foveal. El número de anillos parte de fijar a $\rho_0 = 10$ y sólo se han tomado los primeros 7 radios o factores ρ_ξ calculados.

Tabla 4.1: Indexado de los anillos en la imagen cartesiana a los anillos de la imagen foveal

ξ	1	2	3	4	5	6	7
ρ_ξ	10.1	10.2	10.8	11.3	11.8	12.6	15.4
τ_ξ	1	1	1	2	2	3	3

La variable $\hat{\rho}_\xi$ puede ser calculada fácilmente mediante alguna rutina que se programe previamente al uso de las rutinas de foveación y foveación inversa.

Al reasignar el indexamiento de los anillos cuadrados se debe tomar en cuenta que entonces, un anillo cuadrado de la imagen foveal ya no corresponderá al índice de un anillo cuadrado colocado en la imagen cartesiana. Para conocer el índice del anillo correspondiente, se puede utilizar la siguiente relación:

$$\hat{\xi}_\tau = \underset{\{\xi|\xi \in \{1 \dots N_r\} \wedge \tau_\xi = \tau\}}{\operatorname{argmax}} \rho_\xi, \quad \tau = 0 \dots RN_r \quad (4.2.3)$$

De este modo, los valores de proporción se ajustarán de acuerdo a los anillos que realmente se colocan sobre la imagen foveal. El uso de los factores definidos anteriormente se muestra en los métodos de foveación y foveación inversa explicados a continuación.

4.2.1. Foveación con el método propuesto

Para realizar el mapeo de una coordenada Cartesiana (x, y) a una coordenada foveal (x_f, y_f) , de una imagen foveal como la que se ha propuesto, primero se supone que el muestreo se va a realizar con un número N_r de anillos, el radio del área foveal es ρ_0 , el centro de la región foveal se encuentra colocado en la coordenada (x_0, y_0) y el centro de la imagen foveal se encuentra dado por (x_c, y_c) . Posteriormente, se procede a realizar los siguientes pasos:

1. *Obtener la coordenada de mayor valor.* Esto se usa para saber el radio del anillo cuadrado al que pertenece la coordenada

$$\rho = \text{máx}(|x - x_0|, |y - y_0|)$$

2. *Obtener el número de anillo* Se utiliza la ecuación 4.1.3 del modelo discreto de la TLP ya que esta establece un orden en cuanto a los anillos. También debe recordarse que los anillos se colocan con una distancia logarítmica por lo que entre cada anillo existe una región de pixeles. Esta región se asocia con un anillo de muestreo dependiendo de su proximidad. En términos matemáticos, esto se resuelve al truncar o redondear el valor de la siguiente ecuación.

$$\xi = \log_a \left(\frac{\rho}{\rho_0} \right)$$

3. *Obtener el radio correspondiente al número de anillo.*

$$\rho_\xi = \rho_0 a^\xi$$

4. *Se obtiene la proporción correspondiente.*

$$\Delta\rho_\xi = \frac{\rho_0 + \tau_\xi}{\rho_\xi}$$

5. *Mapeo hacia el plano foveal* La coordenada Cartesiana se divide por la proporción obtenida y se suma el centro de la imagen foveal. Este paso es muy intuitivo

ya que se trata de una regla de proporciones básica. Se suma el centro para colocarse de manera adecuada con respecto al centro de la imagen foveal.

$$\begin{aligned}x_f &= x\Delta\rho_\xi + x_c \\y_f &= y\Delta\rho_\xi + y_c\end{aligned}$$

Los pasos anteriores se cumplen para $\rho > \rho_0$ ya que de otro modo, el mapeo es directo, esto es: $(x, y) = (x_f, y_f)$.

4.2.2. Foveación inversa con el método propuesto

La foveación inversa se refiere a la transformación o mapeo inverso que se aplica para recuperar la imagen original a partir de la imagen foveal. Esta operación inversa es muy importante ya que dado un punto en la imagen foveal (x_f, y_f) , se deseará conocer, para ciertos propósitos, su correspondiente coordenada Cartesiana (x, y) .

El mapeo inverso es similar al mapeo descrito anteriormente y se procede como se explica a continuación:

1. *Obtener la coordenada de mayor valor.*

$$\rho = \text{máx}(|x_f - x_c|, |y_f - y_c|)$$

2. *Obtener el número de anillo* En este caso el número de anillo se obtiene al indexar correctamente el número de anillo foveal con el correspondiente en la imagen cartesiana utilizando la ecuación 4.2.3 como sigue:

$$\begin{aligned}\tau &= \rho - \rho_0 \\ \xi &= \hat{\xi}_\tau\end{aligned}$$

3. *Obtener el radio correspondiente al número de anillo.* Se obtiene el radio en la imagen Cartesiana que corresponde al número de anillo dado.

$$\rho_\xi = \rho_0 a^\xi$$

4. *Se obtiene la proporción correspondiente.*

$$\Delta\rho_\tau = \frac{\rho_\xi}{\rho_0 + \tau}$$

5. *Mapeo hacia el plano Cartesiano.* Se multiplica la coordenada foveal por la proporción obtenida en el paso anterior y se suma el centro en donde se ha colocado la región foveal sobre la imagen Original Cartesiana. Esta última operación permite realizar la foveación móvil o desplazada a cualquier posición en la imagen Cartesiana.

$$\begin{aligned} x &= x_f \Delta\rho_\tau + x_0 \\ y &= y_f \Delta\rho_\tau + y_0 \end{aligned}$$

Del mismo modo que con el proceso de foveación, el mapeo inverso se aplica si se cumple que $\rho > \rho_0$, de otro modo, el mapeo inverso es directo significando que $(x_f, y_f) = (x, y)$.

La figura 4.2 muestra el muestreo foveal bajo el método propuesto de anillos cuadrados con muestreo variable en cada anillo. Estos anillos son colocados sobre la imagen digital como lo muestra la segunda imagen (de izquierda a derecha) en la figura. La tercer imagen muestra la imagen foveal resultado del muestreo. La cuarta imagen muestra la recuperación de la imagen original (primer imagen) mediante la transformación inversa. En esta imagen recuperada se puede apreciar la resolución variante con alta resolución al centro y decreciente hacia las orillas. Incluso, puede comprobarse la existencia de zonas con el mismo valor debido a la correspondencia uno a muchos que se deriva del muestreo.

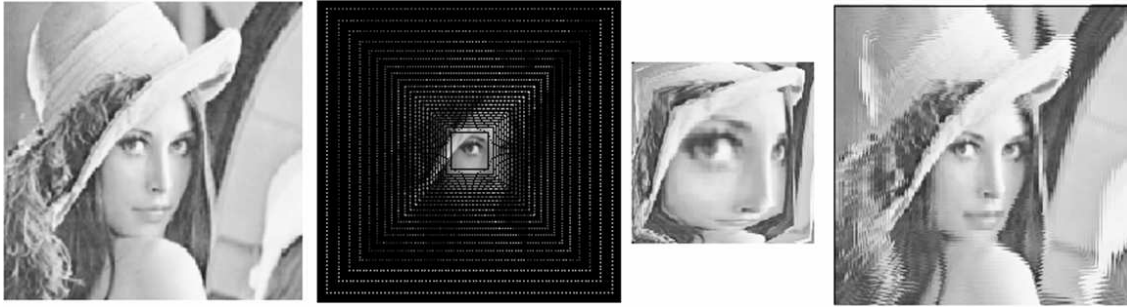


Figura 4.2: Imagen Original, imagen que muestra el muestreo foveal basado en anillos cuadrados, imagen resultado de la foveación e imagen recuperada a partir de la imagen foveal

4.3. Algoritmos de foveación simple y promedio

Los pasos descritos anteriormente permiten diseñar un algoritmo de foveación o muestreo para transformar imágenes digitales convencionales a imágenes digitales foveales. Para el diseño, es importante definir el modo en que se va a muestrear la información ya que, igual a lo que ocurre con el modelo discreto de la TLP, la propuesta presentada de foveación produce una relación de muchos a uno lo que implica que más de una coordenada Cartesiana se mapea a la misma coordenada foveal. Por ello, se proponen dos algoritmos que tienen como paso previo los siguientes cálculos.

1. N : Filas de la imagen Cartesiana
2. M : Columnas de la imagen Cartesiana
3. $ImgC$: Imagen Cartesiana
4. $ImgF$: Imagen Foveal
5. N_r : Número de Anillos
6. ρ_{max} : Radio máximo del anillo cuadrado
7. ρ_0 : Radio de la región foveal
8. (x_0, y_0) : Centro de la imagen Cartesiana

9. (x_c, y_c) : Centro de la imagen foveal
10. $a \leftarrow \exp\left(\frac{1}{N_r} \ln\left(\frac{\rho_{max}}{\rho_0}\right)\right)$
11. $\Delta\rho(\xi) = \frac{\rho_0 + \tau\xi}{\rho_\xi}, \quad \xi \leftarrow 0 \dots N_r$
12. $NF \leftarrow 2(\rho_0 + N_r) + 1$: Filas y Columnas Totales de la imagen Foveal

4.3.1. Foveación simple

Se establece que a cada coordenada foveal (x_f, y_f) le corresponde una coordenada y sólo una coordenada Cartesiana (x, y) . ¿Cómo asignar esta unicidad cuando se tiene una relación de muchos a uno? La respuesta es la misma que para el modelo discreto de la TLP: para cada valor posible de las coordenadas foveales (x_f, y_f) se aplica el método inverso de foveación descrito en la sección anterior. La coordenada cartesiana obtenida se vuelve única al truncar o redondear el paso 5 del mapeo inverso. Para simplificar, a las imágenes foveales generadas con este método se les referirá como imágenes foveales simples.

El algoritmo es el siguiente:

1. Para cada $x_f \leftarrow 0 \dots NF$
2. Para cada $y_f \leftarrow 0 \dots NF$
3. $\rho \leftarrow \text{máx}(|x_f - x_c|, |y_f - y_c|)$
4. $\xi \leftarrow \rho - \rho_0$
5. $x \leftarrow x_f \Delta\rho(\xi) + x_0$
6. $y \leftarrow y_f \Delta\rho(\xi) + y_0$
7. $ImgF(x_f, y_f) \leftarrow ImgC(x, y)$
8. termina
9. termina

4.3.2. Foveación promedio

Este método utiliza el proceso de foveación como tal para cada pixel en coordenadas Cartesianas. Los valores del pixel ubicado en (x, y) son acumulados en un arreglo matricial P en la posición determinada por sus coordenadas foveales. Al final se divide entre el número total de coordenadas Cartesianas C que coincidieron en la coordenada foveal. Para simplificar, a las imágenes foveales generadas con este método se les referirá como imágenes foveales promedio.

1. P : Inicializado a 0
2. C : Inicializado a 0
3. Para cada $x \leftarrow 0 \dots N$
4. Para cada $y \leftarrow 0 \dots M$
5. $\rho \leftarrow \text{máx}(|x - x_0|, |y - y_0|)$
6. $\xi \leftarrow \log_a \left(\frac{\rho}{\rho_0} \right)$
7. $x_f \leftarrow x \Delta \rho(\xi) + x_c$
8. $y_f \leftarrow y \Delta \rho(\xi) + y_c$
9. $P(x_f, y_f) \leftarrow \text{Img}C(x, y)$
10. $C(x_f, y_f) \leftarrow C(x_f, y_f) + 1$
11. termina
12. termina
13. Para cada $x_f \leftarrow 0 \dots NF$
14. Para cada $y_f \leftarrow 0 \dots NF$
15. $\text{Img}F(x_f, y_f) \leftarrow P(x_f, y_f) / C(x_f, y_f)$
16. termina
17. termina

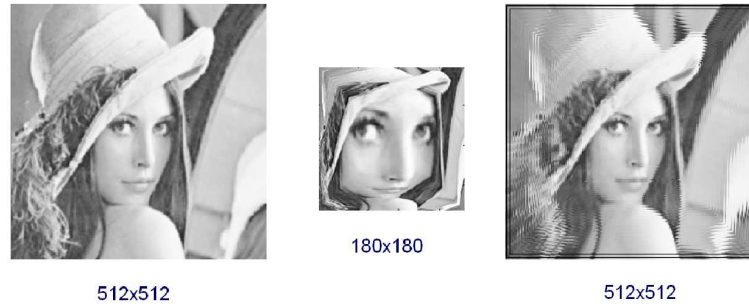


Figura 4.3: Ejemplo del proceso de foveación propuesto, a la izquierda la imagen original a muestrear, en el centro la imagen foveal y en la derecha, la imagen recuperada con la transformación inversa de la foveación

4.3.3. Foveación inversa y recuperación de la imagen

Los algoritmos anteriores son utilizados en este trabajo para llevar a cabo el proceso de foveación. La selección del más adecuado para el seguimiento se discutirá en los siguientes capítulos.

El algoritmo de recuperación es muy sencillo y consiste en reproducir el algoritmo de foveación simple con la diferencia de que la asignación en la línea 7 se invierte tomando en cuenta que la imagen foveal ya contiene información previamente muestreada. Es posible que se desee implementar algún algoritmo más sofisticado como una interpolación más que un simple rellenado, pero en este trabajo, la recuperación de la imagen se utiliza sólo para fines visuales y de comprobación. La rutina de foveación inversa como se ha descrito en los párrafos anteriores resulta suficiente para la recuperación de la imagen original.

Para concluir este capítulo, la figura 4.3 muestra un ejemplo en general del método de foveación expuesto en este capítulo, así como la recuperación de la imagen original mediante la información contenida en la imagen foveal. Se puede apreciar el efecto de resolución variante con alta resolución al centro y decreciente hacia la periferia.

4.4. Discusión

En este capítulo se explicó el método de foveación propuesto. También se describieron las ecuaciones y los algoritmos de foveación y recuperación de la imagen

original. Un aspecto importante a señalar es que el método de foveación se diseñó con el objetivo de realizar un sub-muestreo basado en los principios foveales de máxima resolución al centro y decreciente hacia la periferia. La transformación basada en anillos cuadrados produce algunas deformaciones en los objetos de la imagen a causa del sub-muestreo de los anillos que son colocados a distancias logarítmicas radiales del centro. Sin embargo, los objetos no se dividen abruptamente y preservan la adyacencia ligeramente alterada por la transformación. La relación de adyacencia es importante ya que esto hace posible que los algoritmos basados en apariencias puedan ser utilizados sin problema alguno tal y como se verá en los siguientes capítulos.

Experimentos y resultados obtenidos

En este capítulo se muestran los experimentos realizados y sus resultados, los cuales sustentan las conclusiones de este trabajo de tesis. Primero se discuten algunos puntos para la configuración del método de foveación propuesto y se discuten algunas de las propiedades de las imágenes foveales bajo estudio. Posteriormente se muestran experimentos en los que se han aplicado los algoritmos de detección de movimiento y seguimiento presentados en el capítulo anterior.

5.1. Configuración para la foveación

La foveación o transformación de una imagen cartesiana a una imagen foveal, por el método que sea, implica una reducción de datos que puede verse como una compresión. La reducción de datos obtenida ofrece ciertas ventajas que una simple reducción no ofrece. Por ejemplo, una forma de comprimir una imagen mediante muestreo puede ser la reducción de la imagen a un tamaño menor. Un escalamiento espacial mediante algún algoritmo adecuado puede hacer que la imagen original se recupere sin embargo, existirán pérdidas notables de información en toda la imagen. La ventaja del muestreo foveal radica en que la configuración del muestreo permite comprimir la información y a su vez conservar una zona con alta definición de manera jerárquica, y es alrededor de ésta zona en donde precisamente se alcanza la mayor resolución, siendo así mismo la de grano más fino por contener mayor información.

Una pregunta básica que debe plantearse es: ¿Cuál es el grado de compresión que se

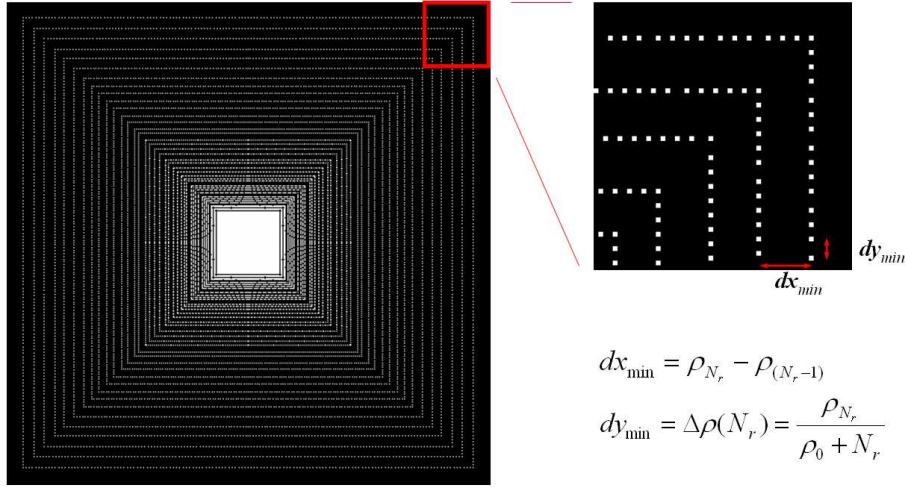


Figura 5.1: Cálculo de las dimensiones del objeto más pequeño que podría ser muestreado por el método de foveación de anillos cuadrados

puede alcanzar con la foveación?. La respuesta no es sencilla ya que el muestreo puede ajustarse tanto como se desee de tal modo que la imagen original quede representada por la menor cantidad de datos posible. No obstante, la cantidad de datos perdida se relaciona con el nivel de visibilidad o representación deseado en cada zona.

De cualquier modo, la foveación como método de compresión no compite con otros métodos bastante efectivos como los utilizados para codificar una imagen en formato JPG o MPEG, ya que el muestreo variante en si puede dejar muchas redundancias en la imagen foveal. De aquí surge la idea de que la foveación puede complementarse con algún algoritmo de compresión para aumentar el factor de compresión.

En [?] se muestran algunos resultados de combinar el efecto de foveación con métodos de compresión. Para ello, el método de foveación se orienta al uso de wavelets y filtros pasa banda. Éste se utiliza para producir el efecto de difuminado variable en una imagen. Para esto se aplica el método de foveación y luego se aplica el método de foveación inversa para recuperar la imagen original pero con los efectos de la degradación no uniforme. Y finalmente a la imagen recuperada se le aplica algún algoritmo de compresión. En este trabajo se muestra también que si la foveación se diseña para producir un factor de compresión de 3 y el algoritmo seleccionado de compresión tiene un factor de 100, al combinarlos se obtendrá un factor de compresión cercano a 300, lo que resulta ideal para transmisión de video.

Para el caso del seguimiento de objetos, y específicamente sobre las imágenes foveales generadas con la propuesta, el factor de compresión quedará determinado por el tamaño mínimo de los objetos que se desean muestrear en las región de los bordes. De manera inversa, dada una imagen foveal, se debe hallar el tamaño mínimo de un objeto que puede ser muestreado. Para responder a esto, obsérvese la figura 5.1 en donde se ha escalado una sección de una imagen con el muestreo foveal. En la imagen escalada se indican dos distancias denominadas como d_x y d_y . Estas distancias indican la separación máxima que hay entre dos puntos de muestreo tanto en x como en y . Esta distancia puede conocerse de acuerdo a las ecuaciones que determinan la distancia de los anillos cuadrados 4.2.1 y el muestreo de cada anillo 4.2.2. Así, para que un objeto tenga una buena representación en el muestreo, es necesario que su ancho y alto sean mayores o iguales d_x y d_y respectivamente.

Tabla 5.1: Parámetros de configuración para el método de foveación

Parámetro	Significado
ρ_0	Radio de la región foveal.
ρ_{max}	Radio máximo del último anillo cuadrado.
N_r	Número de anillos cuadrados.
x_0, y_0	Coordenadas cartesianas donde se colocará el centro de la fovea

El muestreo de los anillos cuadrados puede ajustarse para tomar en cuenta dentro del muestreo a objetos con ciertas dimensiones. El muestreo puede quedar configurado de acuerdo al tipo de escenario y las características de los objetos de interés si se tuviera conocimiento previo de los mismos. Por otro lado, la misma configuración del muestreo podría utilizarse para discriminar objetos que no son de interés. Por ejemplo, en los algoritmos de detección de movimiento un pixel toma un estado binario que indica si en la coordenada asociada al pixel hubo movimiento o no lo hubo. Por lo tanto, la cantidad de movimiento en una región puede determinarse como el conjunto de pixeles adyacentes que presentan movimiento. Si esta cantidad o región de movimiento no cumple con las dimensiones mínimas establecidas - de acuerdo a lo explicado - entonces el muestreo debilitará aún más su presencia. El muestreo foveal entonces puede ser un buen método para realizar un filtrado de información que no se desea considerar en la imagen foveal.

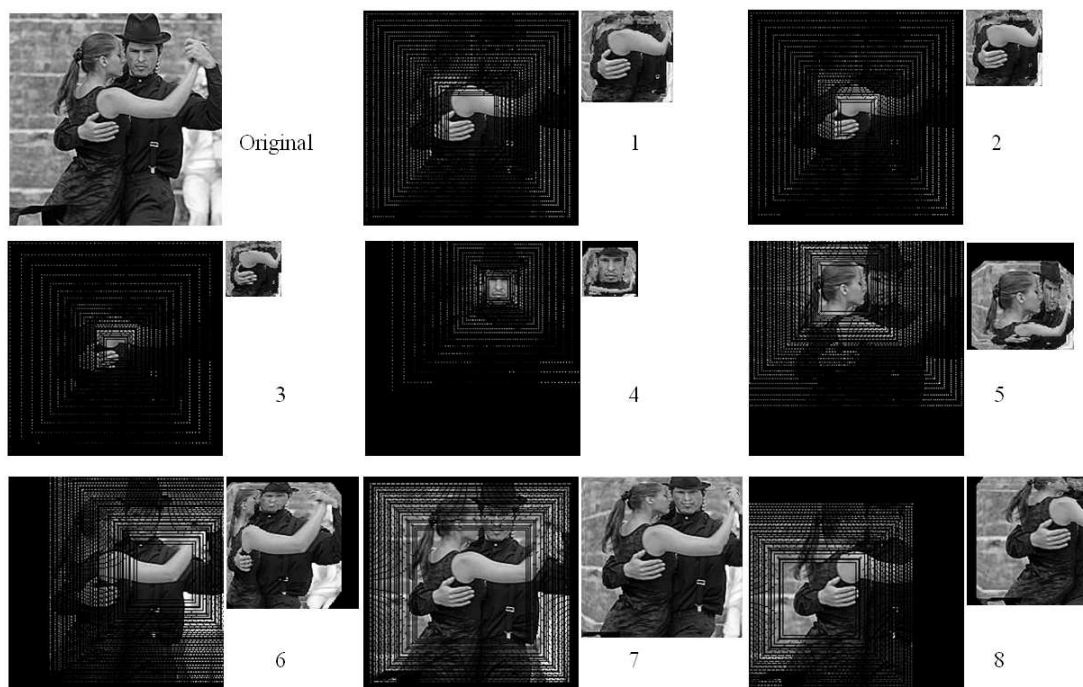


Figura 5.2: Ejemplos de foveación obtenidos de diferentes configuraciones en los parámetros de la tabla 5.1

En la tabla 5.1 se muestran los parámetros que al ser establecidos variarán el modo en el que se lleva a cabo el modelo de muestreo y que se verán reflejados en la imagen foveal de salida. Como se explicó en el capítulo 4, el proceso de muestreo puede adaptarse ya sea al área de máxima resolución o a la región foveal; al número de anillos y por ende al factor de muestreo en cada anillo. Es importante recordar que la distancia a la que se colocan los anillos queda determinada por un factor exponencial en base a y definido en la ecuación 4.1.5. El significado de cada parámetro es el siguiente: **FA** indica el ancho de la imagen foveal resultante, **IA** el ancho de la imagen cartesiana y **Comp** la reducción de datos como la razón de dividir el total de píxeles de la imagen foveal entre el total de píxeles de la imagen cartesiana lo que implica un ahorro substancial de datos.

Por otra parte, el muestreo no necesariamente tiene que aplicarse alrededor del centro de la imagen. Este desplazamiento puede darse al cambiar los parámetros (x_0, y_0) . Esto permite trabajar con una de las restricciones para realizar seguimiento con imágenes foveales y que es mantener la región foveal sobre el objeto de interés.

Sobre todo cuando la cámara es fija y no hay más solución que desplazar el foco del muestreo sobre una coordenada determinada en la imagen cartesiana. La figura 5.2 muestra ejemplos de diferentes modelos de muestreo obtenidos al considerarse diferentes juegos de parámetros de la tabla 5.1. Por otro lado en la tabla 5.2 se muestra que parámetros se eligieron para cada imagen. Se puede observar que el número real de anillos cambia por la discretización ocurrida en la implementación y de acuerdo a lo explicado en la sección 4.2

Tabla 5.2: Parámetros de configuración para las imágenes de la figura 5.2

	x_0	y_0	ρ_{max}	ρ_0	N_r	RN_r	FA	IA	Comp.
1	112	112	112	10	40	38	96	224	0.18
2	112	112	112	5	40	35	80	224	0.12
3	112	112	112	3	30	26	58	224	0.067
4	138	49	112	3	30	26	58	224	0.067
5	103	43	130	10	50	46	112	224	0.25
6	16	203	120	15	60	54	138	224	0.379
7	112	112	112	15	90	69	168	224	0.56
8	73	127	112	20	50	47	134	224	0.35

5.2. Foveación simple y promedio

En las secciones 4.3.1 y 4.3.2 se presentaron dos algoritmos para llevar a cabo el método de foveación propuesto: *La foveación simple y la foveación promedio*. La diferencia substancial entre ellas depende del modo en el que la información es muestreada para construir la imagen foveal.

Cuando se utiliza la foveación simple, en cada punto de la imagen foveal se almacena un valor correspondiente y único de la imagen cartesiana. Este muestreo tiene el inconveniente de dejar fuera del muestreo información de los píxeles vecinos que se encuentran a su alrededor y que no son muestreados. Se puede dar el caso que estos píxeles pueden contener información importante que será eliminada al no ser tomados en cuenta en el muestreo.

En el caso de la foveación promedio, todos los píxeles que coinciden en la misma coordenada foveal son tomados en cuenta para el muestreo. En este trabajo se eligió el

promedio como una alternativa sencilla para involucrar en el muestreo información de dicho píxeles. Otra opción podría haber sido utilizar la mediana o algún filtro de convolución con una máscara centrada y simétrica para dar un peso alto a la contribución de cada píxel (central) involucrado. Lo cierto es que intuitivamente las imágenes foveales obtenidas por el método de promediado o alguno similar, deben proveer más información del escenario sobre todo en las regiones más alejadas de la periferia donde el muestreo decae considerablemente.

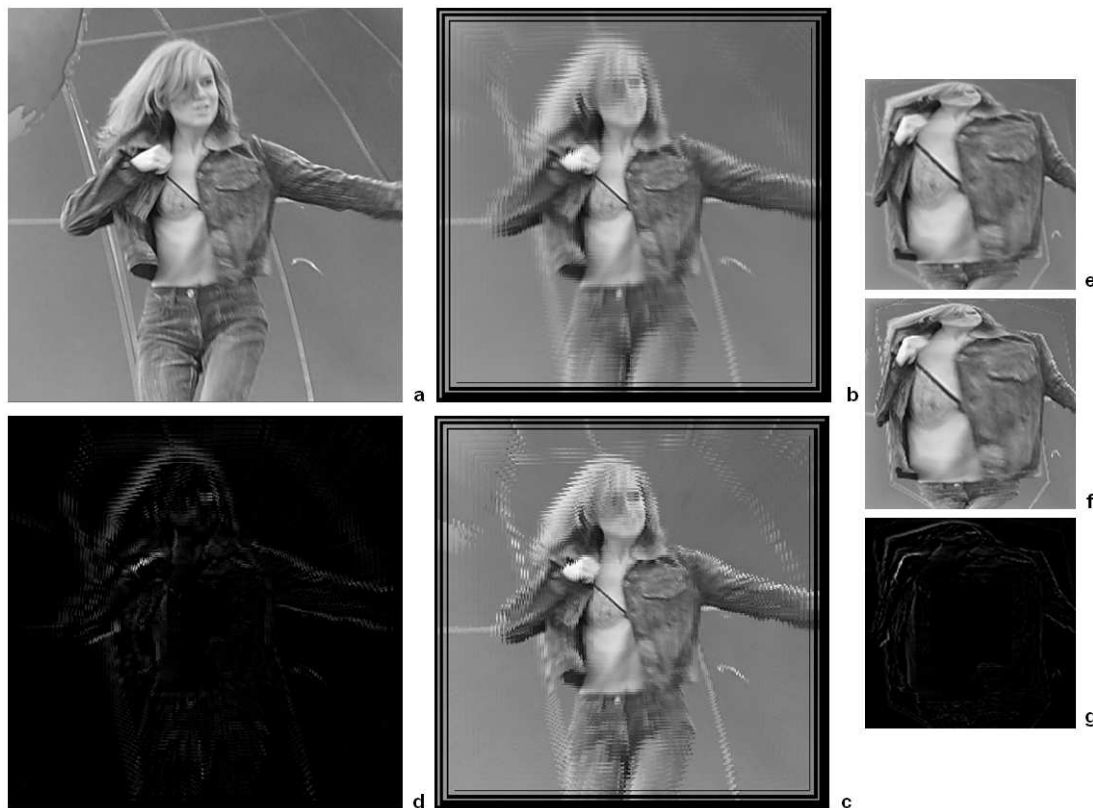


Figura 5.3: Comparación de las imágenes foveales generadas mediante la foveación promedio y la foveación simple

Como se discutió al principio de este capítulo, si un objeto se localiza en las orillas del campo de visión y su tamaño es apenas mayor que el área mínima muestreada por los últimos anillos, entonces su presencia e incluso su dinámica puede pasar desapercibida si se aplica un muestreo simple. En el caso del promediado, se esperaría tener un poco más de información lo que ayudaría notablemente a su detección.

La figura 5.3 muestra la diferencia entre dos imágenes foveales de la misma imagen cartesiana **a** generadas por la foveación promedio **e** y la foveación simple **f**. En la imagen foveal promedio se presenta un efecto de difuminado propio de un filtro tipo media, mientras que la imagen foveal simple se puede apreciar un efecto de agudización. El efecto de promediado da la apariencia de ensanchar la información contenida en la imagen foveal y esto mismo se puede apreciar en la recuperación de la imagen original para cada caso e ilustrado en las imágenes **b** para la promedio y **c** para la simple de la figura 5.3.

La diferencia (pixel a pixel) de imágenes aplicada a las imágenes foveales generadas **g** y las imágenes recuperadas **d** revela este ensanchamiento en la imagen foveal promedio. Este efecto se produce a causa de que varios pixeles en la imagen foveal promedio pueden compartir un mismo valor promedio.

Lo que se ha explicado en esta sección lleva a las siguientes preguntas: ¿en qué casos se debe utilizar la foveación simple y la promedio? ¿es posible trabajar sólo con la foveación simple? ¿cuáles son las ventajas de una y otra? Para contestar a estas preguntas, las siguientes secciones muestran los experimentos realizados en este trabajo de tesis y que giran alrededor del tema de trabajo que es *el seguimiento de objetos en la modalidad de un sólo objeto*.

5.3. Detección de movimiento

En la tabla 5.2, en la última columna (Comp), se puede observar que la reducción de datos ofrecida por las imágenes foveales es substancial. Nuevamente, las ventajas de procesar pocos datos, preservando zonas de interés con calidad de muestreo e información jerárquica del resto del campo de visión, hacen atractivo el uso del muestreo variante. Pero la reducción de datos no es lo único importante; también influye en el desempeño la información que puede ser obtenida con el muestreo de las imágenes foveales. Es decir, ¿cuál es el compromiso de reducir la cantidad de datos en cuanto a la calidad de información que se puede extraer de los mismos?.

Algo que es cierto, es que algunos métodos o técnicas de visión se verán afectadas por la reducción de datos al no contar con la información espacial involucrada en la escena (en el límite de la resolución de adquisición). Sin embargo, otras técnicas

pueden realmente no requerir de toda la información para su desempeño. En esta sección se muestran algunos experimentos en los que se utilizaron imágenes foveales para la detección de movimiento e identificación de blancos de interés. Los algoritmos de detección de movimiento son un bloque importante en la implementación de un sistema de visión por lo que se decidió probar su desempeño en el espacio foveal propuesto.

Para esto se decidió implementar el método de extracción por umbralización mediante el uso del promedio y la desviación estándar, explicado en la sección 3.2.2. Los experimentos consistieron en probar el desempeño de este algoritmo sobre diferentes secuencias características que contenían varios tipos de objetos de interés, objetos deformables (personas, animales), objetos rígidos (automóviles) o la combinación de ambos. El desempeño fue medido como la cantidad de objetos que se detectaron en cada imagen de la secuencia, ya que el algoritmo propone blancos probables producto de la detección, la cual es refinada por un algoritmo de erosión.

Para realizar un etiquetado de blancos se requeriría una etapa siguiente encargada de discriminar el tipo de objetos que son realmente de interés. Para estos experimentos, la erosión elimina bastante ruido y las manchas sobrevivientes generalmente indican el área de los objetos que se encuentran en movimiento sobre la escena o al menos con mayor movimiento que otros. Para estas manchas, se aplica un filtro de área que exige indicar en el escenario aquellos objetos que se movieron y cuya área cubre una cierta cantidad o umbral, eliminando algunas pequeñas manchas de movimiento que pudieran representar ruido. No se descarta que el algoritmo incurra en detectar movimiento de árboles o detecte sombras. Para estas pruebas, es de interés saber cuántas detecciones, una vez realizado el filtrado realizó el algoritmo.

Para hacer la comparación, se aplicó el algoritmo a una secuencia de imágenes cartesianas o de resolución invariante donde se encerraban en cajas mínimas envolventes a los objetos detectados en cada imagen. Posteriormente, se eligió una configuración foveal para generar las imágenes foveales de la secuencia y se aplicó el algoritmo de detección. La tabla 5.3 presenta el nombre de las secuencias (**Secuencia**) utilizadas, así como las características de configuración para él. La información descrita en cada columna indica las dimensiones de las imágenes para cada secuencia (**Dimensiones**), el número de imágenes que componen a la secuencia (**Imágenes**) y los parámetros de configuración del algoritmo de detección de movimiento. De acuer-

do al algoritmo explicado en 3.2.2, se describen los valores de (α) utilizados para cada secuencia, el área mínima (**Área**) que deben cumplir las manchas de movimiento para ser tomadas en cuenta y el número de imágenes que se observan para construir al modelo (**Obs.**). Cabe señalar que los parámetros se eligieron en función del desempeño del algoritmo de tal manera que la detección se realizará con el menor número de falsas detecciones. De cualquier modo, al tratarse de una etapa inicial de detección, no se descartó que algunas detecciones de movimiento de árboles o sombras fueran realizadas sin embargo, el ajuste de estos parámetros se hizo para que fueran mínimas.

Tabla 5.3: Configuración del algoritmo para la detección de movimiento

Secuencia	Dimensiones C	Imágenes	α	Área	Obs.
MeetSplit3rdGuy	384x288	897	5	15x15	25
WalkByShop1cor	384x288	2299	3	30x30	60
Carretera3	768x586	1625	5	40x40	30

5.3.1. Experimentos

Para las pruebas, la configuración de los métodos de foveación y el método específico utilizado se muestra en la tabla 5.4. Para la configuración, el número de anillos y el radio del área foveal se eligieron primero en función del área mínima a detectar (sección 5.1), en este caso incluso el radio del área foveal no es relevante. La segunda y en unos casos hasta la tercera forma de configuración, se eligió de tal forma que la cantidad de datos a procesar en la imagen foveal aumente con respecto a la primera configuración y se acerque a la mitad o a un cuarto de los datos totales a procesar de la imagen original.

Los experimentos consistieron en aplicar el algoritmo de detección de movimiento en ambas secuencias, la secuencia de imágenes cartesianas originales y la secuencia de imágenes foveales obtenidas por la foveación. El algoritmo señala a todo movimiento o región de movimiento en un rectángulo o caja mínima envolvente. Para el caso de las imágenes foveales, la detección se hace sobre la imagen foveal pero la indicación de movimiento se realiza sobre la imagen inversa. De este modo, aún cuando la detección sea en el plano foveal, se contará con las cajas mínimas envolventes que indican movimiento en el plano cartesiano también. Para evaluar el desempeño de la detección

Tabla 5.4: Secuencias de prueba para la detección de movimiento

Secuencia	Foveación	ρ_0	N_r	Correctos	% Corr.	Fov. Dim.	% Reducción
MeetSplit3rdGuy	Promedio	20	60	831	0.926	156x156	0.22
MeetSplit3rdGuy	Simple	20	60	744	0.829	156x156	0.22
MeetSplit3rdGuy	Promedio	20	100	862	0.96	216x216	0.42
MeetSplit3rdGuy	Simple	20	100	831	0.926	216x216	0.42
WalkByShop1cor	Promedio	20	60	2277	0.99	156x156	0.22
WalkByShop1cor	Simple	20	60	2231	0.97	156x156	0.22
WalkByShop1cor	Promedio	20	100	2285	0.99	216x216	0.42
WalkByShop1cor	Simple	20	100	2249	0.978	216x216	0.42
Carretera3	Promedio	30	60	1558	0.958	180x180	0.072
Carretera3	Simple	30	60	1507	0.927	180x180	0.072
Carretera3	Promedio	30	100	1564	0.96	256x256	0.145
Carretera3	Simple	30	100	1525	0.938	256x256	0.145
Carretera3	Promedio	30	200	1580	0.97	402x402	0.359
Carretera3	Simple	30	200	1546	0.95	402x402	0.36

en las imágenes foveales se toman estas cajas, representadas por dos pares de coordenadas (esquina superior izquierda y esquina inferior derecha), y se comparan contra los detectados para esa imagen por el algoritmo aplicado, pero a la imagen cartesiana original. Se contabilizará un acierto cuando el número de objetos detectados en la imagen foveal coincida exactamente con el número de objetos detectados cuando se utiliza la imagen cartesiana original. El porcentaje de aceptados (correctos) resulta de dividir el número de imágenes en las que la detección de regiones con movimiento en la representación cartesiana coincide en número con el detectado en la foveal.

En las siguientes figuras se muestran ejemplos de detección en imágenes cartesianas y foveales de las secuencias antes presentadas. En estas imágenes se podrá observar que las áreas de las cajas mínimas - que indican la detección de movimiento - no tienen exactamente las mismas dimensiones. Esto se debe a que en el plano foveal, al recuperar una coordenada cartesiana, el proceso de redondeo o truncamiento de la salida del método de foveación inverso introducirá un error que se verá reflejado precisamente en un ensanchamiento o reducción de las dimensiones de estos rectángulos. En la comparación, para determinar si una región detectada en el marco foveal es la misma que una detectada en el cartesiano, se comprueba si las cajas se intersectan.

Si es así, entonces se contabiliza como detectado.

En la tabla 5.4 la información que se muestra en las columnas indica lo siguiente: la secuencia de trabajo (**Secuencia**), el método de foveación utilizado (promedio o simple) (**Foveación**), el radio de la región foveal \mathbf{p}_0 , el número de anillos N_r , la cantidad de imágenes en donde la cantidad de regiones detectadas en el plano cartesiano coincide con las detectadas en el foveal (**Correctos**), el porcentaje respectivo con respecto al total de imágenes de la secuencia (**% Corr.**), las dimensiones de la imagen foveal generada por la configuración (**Fov. Dim.**), y el porcentaje de datos a los que se reduce el espacio de trabajo para el algoritmo, al procesar las imágenes foveales.

Secuencia MeetSplit3rdGuy

Esta secuencia muestra un escenario interior donde algunas personas caminan a lo largo del escenario. Las personas entran al escenario de diferentes direcciones, estas se reúnen al centro y posteriormente se dispersan en direcciones contrarias formando grupos de dos o de una persona. Esta secuencia ha sido tomada de la base de datos del proyecto CAVIAR [?].

La secuencia resulta interesante porque las personas o grupos de personas caminan a lo largo de todo el salón, entrando y saliendo de las orillas de las imágenes. Como puede apreciarse en los resultados de la tabla 5.4, los porcentajes en el campo de *Correctos* alcanzan a colocarse por arriba del 90 %. De estos datos se puede apreciar que las imágenes foveales obtenidas con el método de foveación promedio presentan los porcentajes más altos en la detección e incluso, las imágenes foveales generadas con el método simple de dimensiones 156×156 son las que presentan el porcentaje más bajo. Lo anterior muestra la utilidad de incorporar en el muestreo información de todos los pixeles que coinciden en una misma coordenada foveal.

Algo importante a observar es que el porcentaje de *Correctos* no necesariamente tiene que llegar al 100 % puesto que la imagen foveal es un subconjunto de los datos de la imagen original y por lo tanto, como se ha discutido, la imagen foveal filtra información que no alcanza a tener una área de movimiento lo suficientemente apreciable como para ser muestreada por los anillos cuadrados.

El porcentaje de imágenes en las que no coincidió el número de regiones de

movimiento detectadas fue revisado. En estos casos se encontró que efectivamente, las regiones de movimiento que no se detectaron fueron aquellas que partían de las orillas y cuyas áreas de movimiento no podían ser percibidas por el muestreo. En cuadros posteriores los objetos que producen las regiones de movimiento avanzan hacia el centro de la imagen y son detectados gracias a que la distancia disminuye entre los puntos de muestreo en la imagen foveal. De hecho, se puede apreciar que en las imágenes foveales simples de dimensiones 216×216 se da un incremento en el porcentaje de *Correctos* debido a que el área mínima de detección aumenta. Esto resulta lógico al tomar en cuenta que entre más anillos mayor será la capacidad de detección de las imágenes foveales.

Otro punto interesante es la reducción de datos (factor de compresión) que se tiene al trabajar con las imágenes foveales, la cual es mucho menor a la mitad de la cantidad de datos originales. Aquí se refleja la utilidad complementaria de la visión foveal, ya que la detección lograda en las imágenes foveales es alta y se logra con una porción mucho menor de datos, propiedades que pueden ser aprovechadas para trabajar bajo restricciones de tiempo real con aquellos algoritmos que demandan un proceso elaborado de procesamiento pixel a pixel.

La figura 5.4 muestra dos ejemplos de imágenes analizadas de la secuencia *Meet-Split3rdGuy*. En la primera serie de imágenes se muestra la detección de movimiento cuando se procesa sobre las imágenes cartesianas. La segunda serie muestra las imágenes cartesianas recuperadas de las imágenes foveales donde se señalan las regiones de movimiento detectadas en las imágenes foveales. La tercera serie de imágenes corresponde a las imágenes foveales, en donde se observa la detección de movimiento y es señalada en las imágenes recuperadas. Para ambos ejemplos se puede observar que el número de regiones coincide en ambos casos, tanto los detectados en las imágenes cartesianas, como en las foveales. En ambos casos se señala con amarillo la región de movimiento de mayor área rectangular. Para el caso de las imágenes foveales esto es sencillo ya que los rectángulos son recuperados desde coordenadas cartesianas y entonces es fácil determinar cuál tiene mayor área.

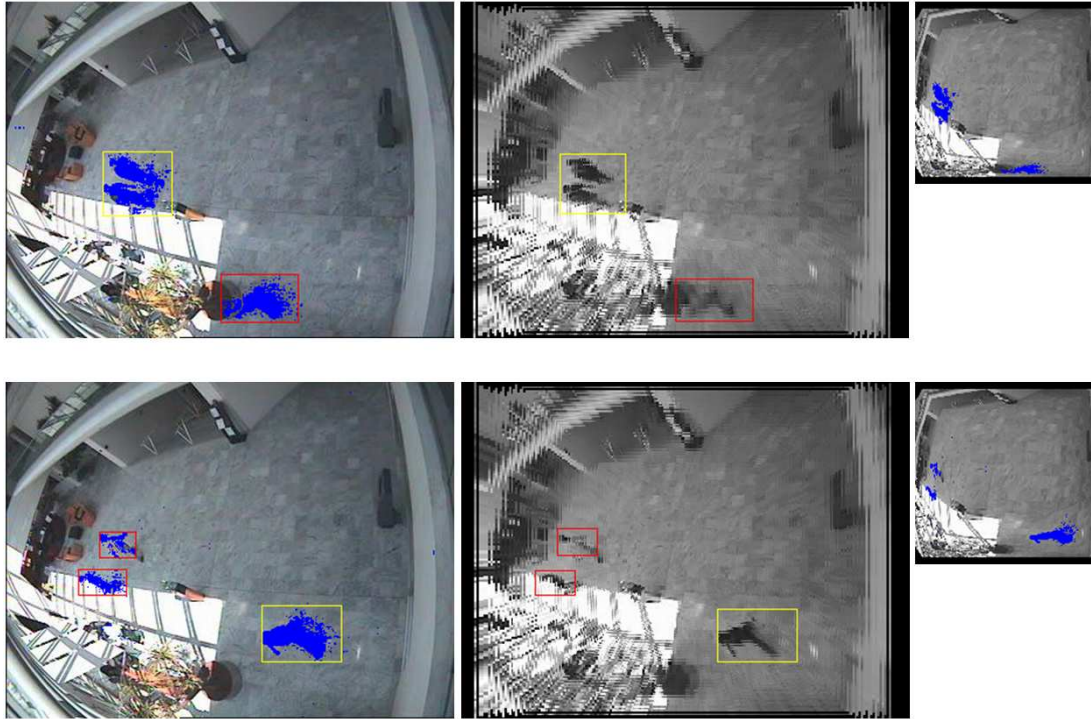


Figura 5.4: Detección de movimiento en la secuencia MeetSplit3rdGuy Cartesiana y foveal

Secuencia WalkByShop1cor

Esta es una secuencia tomada también de la base de imágenes del proyecto CAVIAR [?]. En este caso, la vista es frontal y no superior como en la secuencia anterior. Contiene un gran número de imágenes y en ellas se muestra a varias personas caminando a la largo de un pasillo, caminando de extremo a extremo. Esta secuencia resultó de interés ya que tomando las mismas configuraciones para la foveación que en la secuencia anterior, se obtuvieron porcentajes en el campo de *Correctos* arriba del 90% y en unos casos muy cercano al 100%, gracias a que muchas de las regiones de movimiento son lo suficientemente grandes para ser detectadas por el muestreo. Como en el caso anterior, la detección en imágenes foveales promedio es mejor que la detección simple pero cabe señalar que la detección simple tiene un mejor desempeño que en la secuencia anterior lo que indica que, con una configuración adecuada, es posible obtener buenos desempeños en la detección para imágenes foveales simples junto a la ventaja de procesar menores cantidades de información de manera significativa.

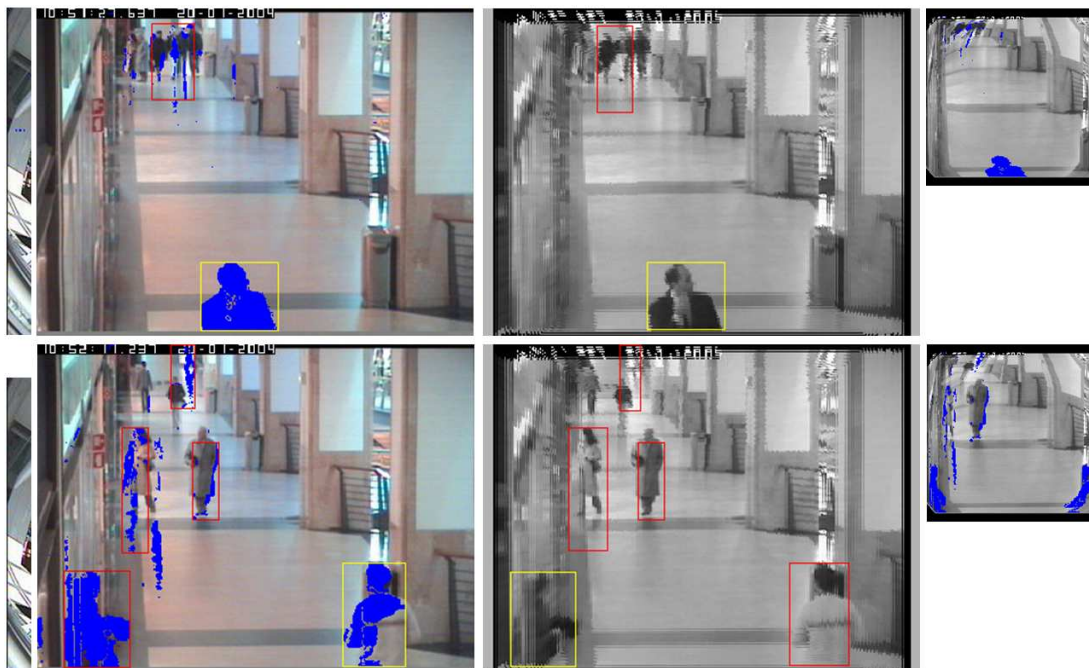


Figura 5.5: Detección de movimiento en la secuencia WalkbyShop1cor en Cartesiano y foveal

En la figura 5.5 se muestra un par de imágenes de la secuencia que fueron analizadas y en las que se muestra la detección tanto en las imágenes cartesianas como en las foveales, en éstas la detección de la región es marcada sobre las imágenes recuperadas.

Secuencia Carretera

Esta es una secuencia de un escenario exterior en donde transitan personas, automóviles y donde hay movimiento de árboles. Al igual que en los ejemplos anteriores, se obtuvieron las imágenes foveales correspondientes mediante diferentes configuraciones en el método de foveación y utilizando ambos tipos de foveación: promedio y simple. Para este caso se probaron tres tipos de configuraciones en donde el porcentaje de *Correctos* se también se mantiene por encima del 90 % con una reducción significativa en los datos a procesar.

Esta secuencia se utilizó para mostrar que la foveación no sólo es útil para realizar detección a bajo costo respecto a la cantidad de datos a procesar y por lo tanto, en tiempo de procesamiento; sino también para mostrar que la detección jerárquica es

aplicable.

En la figura 5.6 en la primer fila de imágenes, se muestra un ejemplo en donde la detección en la imagen foveal no indicó una región que sí fue detectada en la imagen cartesiana. Esto se debe probablemente a que el movimiento presentado es pequeño (el balanceo del árbol) pero suficiente para sobrevivir a la filtración de erosión y área en la imagen cartesiana, pero en el caso de la foveal, esto ya no es posible puesto que su ubicación se encuentra en la periferia, en donde su área no alcanza a ser suficientemente muestreada.

5.3.2. Discusión

La detección de movimiento utilizando imágenes foveales muestra una ganancia en los costos de procesamiento reflejados en el ahorro de tiempo, lo que es importante para aplicaciones en tiempo real. Posiblemente, los resultados anteriores podrían haberse obtenido de manera similar utilizando imágenes log polares. Sin embargo, la ventaja de utilizar las imágenes foveales generadas con el método propuesto en este trabajo permiten una ventaja adicional que es la linealidad y preservación de la adyacencia de los pixeles, lo que permite un análisis más simple de la dirección del movimiento. En el caso de las imágenes log polares, este análisis es más arduo [?] a causa de la no linealidad. La detección de movimiento en la pirámide de imágenes obtenidas por las geometrías cartesianas exponenciales también es lineal, pero el problema se origina ahora en el filtrado de las regiones detectadas. Por cada nivel o imagen de la pirámide se debe configurar el filtro de erosión o de discriminación por área para determinar qué regiones serán tomadas como movimiento y cuáles no. Una vez realizada la discriminación, se deberá analizar qué regiones coinciden porque es claro que lo que se detecte en un nivel también se detectará en un nivel siguiente de mayor resolución, por lo que habrá buscar una manera de asociar esta información, sobre todo si en un nivel el área se marca como no movimiento a causa de la baja resolución, y en otro nivel de mayor resolución la misma región si se marca. Como puede imaginarse, la detección de movimiento en las geometrías exponenciales cartesianas exige un algoritmo más elaborado.

Por otra parte, la detección de movimiento en las imágenes foveales propuestas

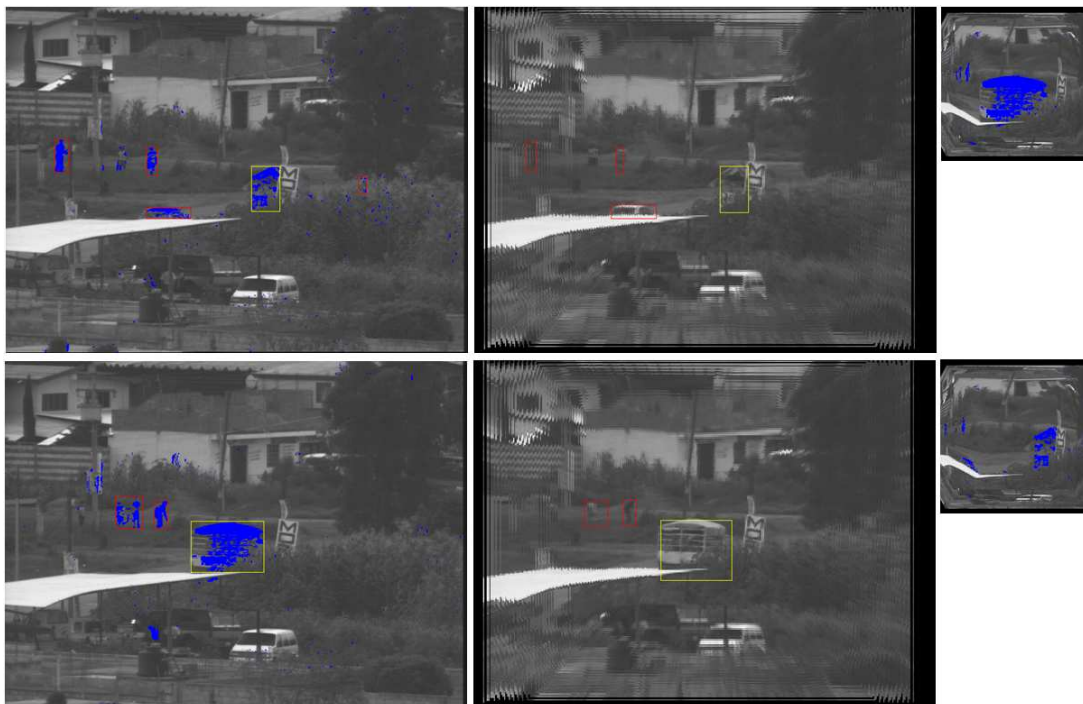


Figura 5.6: Detección de movimiento en la secuencia Carretera en Cartesiano y foveal

puede ser complementada con un módulo de seguimiento que elija algún blanco en las regiones donde se detecte movimiento. En este trabajo, el blanco se identifica como la región de movimiento con mayor área rectangular (rectángulo amarillo). En una aplicación de vigilancia esta región puede comenzar a ser detectada y seguida por un módulo de seguimiento ad hoc con un enganche del objeto que produce este movimiento, ya sea que se indique manualmente o mediante alguna heurística (si el rectángulo aparece en un radio de cierta cantidad de pixeles a lo largo de un número determinado de imágenes consecutivas, entonces enganchará la región detectada).

5.4. Seguimiento de un objeto en la escena

El seguimiento es el problema principal a resolver en este trabajo. En las secciones anteriores se ha mostrado una de las grandes ventajas de utilizar imágenes foveales para una tarea de visión complementaria en el seguimiento y ésta es la detección

de movimiento. En esta sección se presentan los experimentos realizados para llevar a cabo el seguimiento utilizando las imágenes foveales generadas con el método propuesto. El algoritmo utilizado es el expuesto en la sección 3.6.4 que se basa en el seguimiento natural con los estados de fijación y de movimiento sacádico.



Figura 5.7: Seguimiento de un objeto sobre la imagen foveal con fovea fija

5.4.1. Seguimiento con fovea fija

Antes de mostrar estos experimentos, se muestran primero un ejemplo de seguimiento convencional pero aplicado a la imagen foveal con región o zona foveal fija. Esto es, la región foveal no se mantiene sobre el objeto de interés y por lo tanto el seguimiento se aplica como si se tratara de una imagen cartesiana convencional. El objetivo es mostrar la ventaja de la linealidad existente en las imágenes foveales propuestas en este trabajo.

La primera ventaja es que un algoritmo basado en apariencias puede ser utilizado en toda la imagen foveal, incluyendo la periferia y la región foveal. La segunda es que la trayectoria no sufre saltos abruptos o cambios de dirección no continuos. Aunque la detección comienza a fallar cuando el objeto se aleja del centro hacia la orilla, el ejercicio permite mostrar que la detección de blancos puede llevarse a cabo bajo ciertos límites. La figura 5.7 muestra una imagen con el movimiento superpuesto de un objeto que atraviesa el escenario de derecha a izquierda y que se muestra en la imagen foveal colocada primero en la tercer columna de imágenes. En las primeras dos imágenes, la imagen original y la recuperada, también se han superpuesto las

trayectorias del objeto junto con la detección obtenida por la medida de similitud o el módulo de detección de movimiento según haya sido el caso. La detección señalada en la imagen cartesiana es la misma que la señalada en la imagen recuperada. La detección se muestra en ambas para apreciar la pérdida de datos en la periferia sobre la imagen recuperada con respecto a la original.

La figura 5.8 muestra el error en posición del seguimiento del objeto en las imágenes foveales generadas con ambos métodos, con el método de foveación promedio y el método de foveación simple. Debe notarse que las imágenes foveales promedio introducen ruido en la detección, lo cual es lógico debido a que la coordenada que reporta la detección es el centro de la plantilla o subimagen en donde se encontró el valor de similitud más alto. En este caso, es el centro del rectángulo rojo. Para calcular el error de la posición se grafica por cada imagen la distancia euclidiana de la posición reportada por el algoritmo de seguimiento con respecto a la posición verdadera del objeto, la cual se conoce anticipadamente. La detección y en general el seguimiento falla cuando el objeto se ha hecho muy pequeño al retirarse hacia la orilla, dificultando la detección por la medida de similitud o el módulo de detección de movimiento. El algoritmo utilizado para el seguimiento se complementó con una estrategia de actualización de la plantilla en cada imagen. esto se hizo para enfrentar el cambio de forma del objeto debido al muestreo foveal.

5.4.2. Seguimiento con fovea móvil

En la sección anterior se mostró un ejemplo en el que se lleva a cabo el seguimiento de un objeto sobre la imagen foveal sin mover la fovea, como si se tratara de un seguimiento en una imagen convencional. La gráfica 5.8 muestra que el seguimiento hacia la periferia pierde precisión conforme uno se acerca a la orilla. Esto muestra que las imágenes foveales no deben ser utilizadas como imágenes convencionales para la detección de objetos. Al contrario, las propiedades inherentes de las imágenes foveales deben ser explotadas en aquellas tareas que se ajusten a un seguimiento foveal, significando que un objeto de interés requiere de una región de atención que debe vigilarse con mayor detalle que el resto del escenario y sin dejar de analizar el resto de éste aunque con menor énfasis.

En la sección 5.3 se ha mostrado la utilidad de la zona periférica para detectar

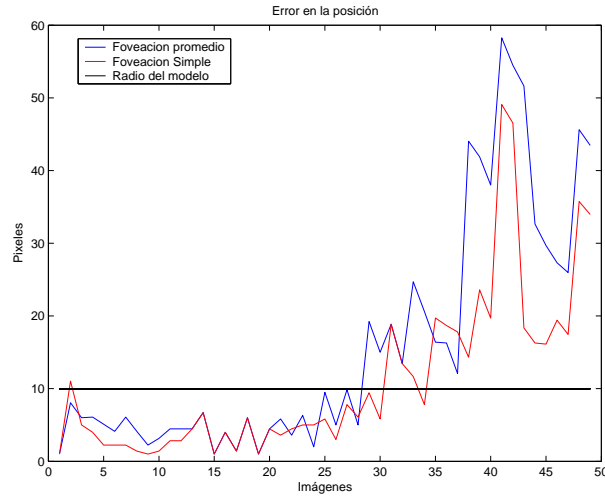


Figura 5.8: Comparación del error de posición del seguimiento sobre imágenes foveales obtenidas con la foveación promedio contra el seguimiento en imágenes foveales obtenidas con la foveación simple

regiones de interés que podrían convertirse en blancos de interés. La detección en esta región como se ha explicado no necesariamente tiene que ser simple, sino que más bien, la simpleza se refiere a que en la zona periférica, dependiendo de la resolución, la información con la que se cuenta es más simple y hay menor detalle de ésta, no impidiéndose así que se puedan utilizar métodos más elaborados para detección de movimiento como lo son: flujo óptico, métodos probabilísticos o métodos de aprendizaje automático, por mencionar algunos. Lo importante aquí es distinguir la funcionalidad de la zona periférica dentro de un sistema de visión, en el caso de los dedicados a la vigilancia se requiere analizar el escenario de manera jerárquica en función de una región de interés, así cuando un objeto de interés sea detectado en la zona periférica y tenga un valor de interés significativo, entonces el sistema deberá mover la zona de atención con mayor resolución hacia dicho objeto, y así lograr un análisis detallado tal como se impone en el seguimiento (medidas de similitud, asociación de datos, mantenimiento de trayectoria, entre otros).

La geometría foveal que se ha propuesto en este trabajo permite mantener la región de mayor resolución sobre el objeto de interés de tal forma que el análisis de los datos pueda realizarse con menor complejidad, respecto al que se exige sobre una imagen convencional y sin perder la relación espacial de adyacencia de los píxeles, lo que permite aplicar los algoritmos convencionales de detección y seguimiento. La

región foveal en las imágenes foveales propuestas equivale a una ventana de búsqueda y se realiza el seguimiento usando a la región foveal como centro, lo cual equivale propiamente a una simple detección sobre una ventana de búsqueda. Sin embargo la idea principal de este trabajo es mezclar las virtudes del seguimiento convencional que funciona exitosamente y las ventanas de búsqueda en el contexto de un modelo con resolución uniforme.

La idea principal realmente consiste en utilizar cada región de la imagen foveal para un objetivo determinado. Esto es, la región foveal, que corresponde a la región o ventana de búsqueda con resolución invariante y lineal, se utiliza para la detección. Por otra parte, la región periférica se utiliza para obtener información jerárquica de interés que pueda resultar importante para el seguimiento.

5.4.3. Experimentos

En base a lo antes discutido los experimentos de seguimiento que se plantearon muestran una situación en la que se mezcla la detección sobre la región foveal y el análisis de la región periférica como complementos para realizar el seguimiento de un objeto de interés. Un escenario difícil para estas pruebas es aquel donde ocurre una oclusión, de tal manera que el objeto se pierda.

El experimento realizado consiste en detectar y mantener a un objeto de interés sobre la región foveal reportando la trayectoria detectada. Cuando la oclusión se presenta, la similaridad reportará un valor por debajo de un umbral de confiabilidad. En este momento, al no saber el número de imágenes que contienen la oclusión, el algoritmo procederá a cambiar del modo de detección al modo de análisis de la región completa (análisis de movimiento) para obtener un blanco probable y desplazar la región foveal hacia dicha región y volver al módulo de detección o en caso contrario continuar con la detección, tal y como se ha detallado en el algoritmo explicado en la sección 3.6.4.

Las secuencias elegidas corresponden a un escenario con fondo fijo en donde un objeto de interés es enganchado. El seguimiento se mantiene hasta que se produce una oclusión o una pérdida momentánea del objeto de interés y por lo tanto se hace la identificación con el módulo de detección de movimiento. Para simular una pérdida abrupta del objeto de interés se saltaron algunas imágenes continuas en la secuencia

con el fin de que el algoritmo trabaje con las diferentes regiones de la imagen foveal (esto equivale a una pérdida de foco).

Por otra parte, el algoritmo de seguimiento foveal tiene una implementación similar para ser aplicado en imágenes cartesianas y la diferencia consiste en desplazar la ventana de búsqueda sobre la detección tal y como se desplaza la región foveal sobre la detección también. Este algoritmo se ha explicado de manera pormenorizada en la sección 3.6.2. Las pruebas también se realizaron sobre las imágenes cartesianas para comprobar el desempeño del algoritmos sobre los diferentes espacios de trabajo.

Las pruebas se realizaron para contrastar utilizando las imágenes foveales producidas por el método de foveación promedio, y posteriormente sobre las producidas por el método simple. El objetivo es observar su desempeño y valorar en que caso vale la pena utilizar un método u otro.

Como medida de error se utiliza nuevamente la distancia euclidiana que hay entre cada punto detectado contra el punto real correspondiente al objeto de interés en la escena (Ground Truth GT).

La tabla 5.5 muestra las configuraciones para el método de foveación usado para cada secuencia. Las columnas indican respectivamente: la secuencia correspondiente (**Sec.**), las dimensiones de las imágenes cartesianas (**Dimen. Sec.**), el número de imágenes que componen la secuencia (**Imágenes**), el radio del modelo o plantilla patrón del objeto de interés (**Radio Modelo**), el radio de la región foveal - que equivale al radio de la ventana de búsqueda cuando se realizan las pruebas en las imágenes cartesianas ($\rho_0 = \mathbf{Radio VB}$)¹, el número de anillos (ρ_0), las dimensiones de la imagen foveal obtenida con dicha configuración (**Dimen. Fovea**) y por último, la reducción de datos que se tiene al trabajar con la imagen foveal respecto a procesar toda la imagen de manera cartesiana (**% Reduc.**).

Secuencia 1

La figura 5.9 muestra un par de ejemplos de seguimiento, realizados sobre las imágenes de la secuencia 1 que muestra una lancha que viaja de izquierda a derecha a lo largo del escenario y pasando por detrás de un poste. Como se explicó en la

¹Este radio se asigna como el doble del radio del modelo de acuerdo a lo discutido en la sección 3.4.

Tabla 5.5: Parámetros de configuración para la foveación en los experimentos realizados

Sec.	Dimen. Sec.	Imágenes	Radio Modelo	$\rho_0 =$ Radio VB	N_r	Dimen. Fovea	% Reduc.
1	640x480	450	12	24	35	118x118	0.045
2	640x480	440	12	24	35	118x118	0.045
3	640x480	74	15	30	35	130x130	0.055
4	480x360	200	15	30	35	130x130	0.0978
5	512x512	514	15	30	35	130x130	0.064

descripción de las pruebas, el seguimiento se realizó con las imágenes cartesianas bajo la estrategia de utilizar una ventana de búsqueda (primer columna de imágenes en la figura) y sobre las imágenes foveales que en todo momento reportaban la coordenada cartesiana correspondiente con la ayuda de la transformación inversa (segunda y tercer columna de la figura).

En la figura 5.10, la imagen de la izquierda muestra el error en la determinación de la posición con respecto a la trayectoria real del objeto. El radio de la plantilla se ha indicado con la recta horizontal negra que atraviesa a la gráfica para indicar el error de la posición de la ventana de búsqueda o la región foveal sobre el objeto de interés, ya que se trata de una zona de atención.

Los instantes (o frames) en los que ocurre la oclusión se reflejan como un pico en las curvas, éste corresponde a una fluctuación inducida por la oclusión. De ésta gráfica de error puede apreciarse que el desempeño del algoritmo en cada prueba es similar. Las variaciones que se tienen entre las curvas ocurren debido a que el objeto sufre algunos cambios en su forma debido al oleaje y a los cambios de luz, por lo que la similitud se ve reducida y el módulo de detección de movimiento entra en acción.

La figura 5.10, imagen de la derecha, muestra el tiempo de ejecución acumulado de los algoritmos en cada espacio de trabajo. El tiempo de foveación, utilizado ya sea para el método promedio o simple, es contabilizado junto con el algoritmo de seguimiento por tratarse de una transformación por software. Claramente la foveación promedio eleva su costo en tiempo debido a que requiere procesar toda la imagen original para obtener el promediado. Este es el inconveniente del método de foveación promedio cuando se aplica por software. Por el contrario, el método simple se presenta como el de menor consumo de recursos en procesamiento: 1) porque el método de foveación

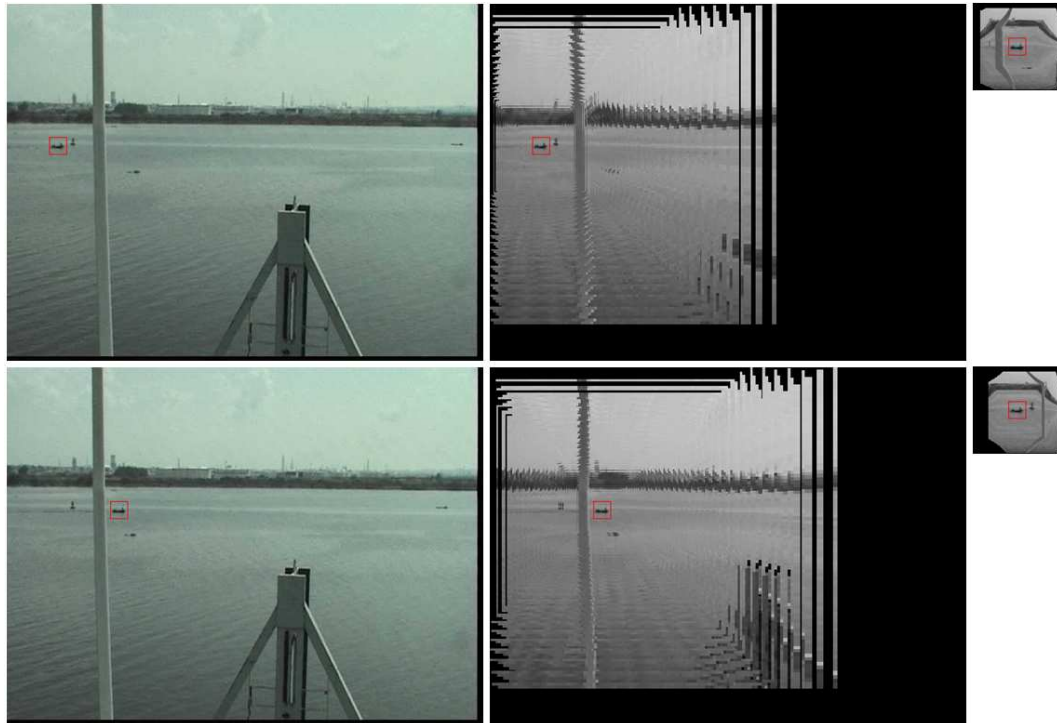


Figura 5.9: Secuencia 1

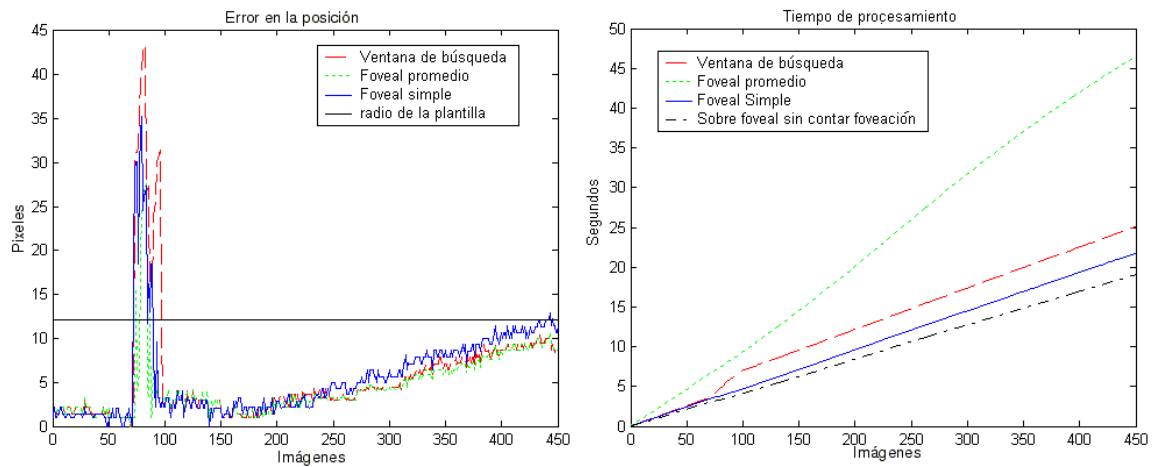


Figura 5.10: Resultados obtenidos para el seguimiento en la secuencia 1

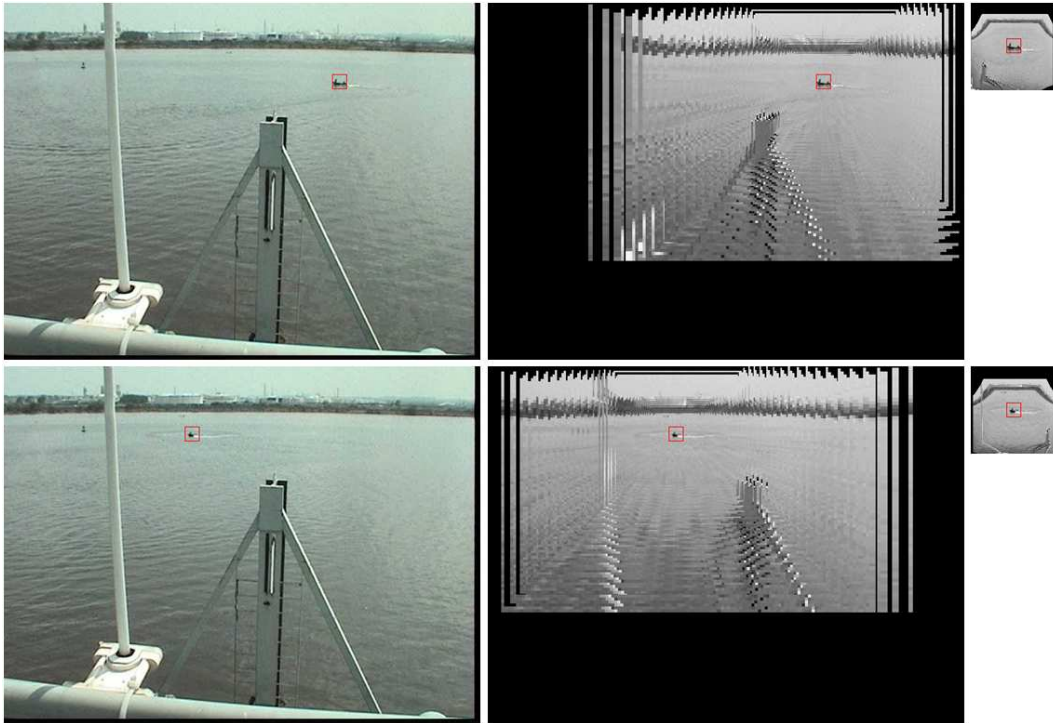


Figura 5.11: Secuencia 2

es rápido y 2) debido a que el módulo de detección de movimiento procesa menos información que el utilizado en imágenes cartesianas.

La curva negra en la gráfica de tiempo muestra la ejecución del algoritmo de seguimiento en el espacio foveal sin tomar en cuenta el tiempo invertido en la foveación. El objetivo es apreciar el ahorro en tiempo de procesamiento debido al hecho de procesar menos datos en la imagen foveal.

En el caso de las imágenes cartesianas, el tiempo comienza a elevarse cerca de la imagen 80 cuando se enfrenta uno a la oclusión. El hecho de procesar toda la imagen para llevar a cabo la detección de movimiento exige mayor tiempo de ejecución. Para los siguientes experimentos, este comportamiento será el que comúnmente eleve los tiempos de ejecución cuando se realice el seguimiento en imágenes cartesianas.

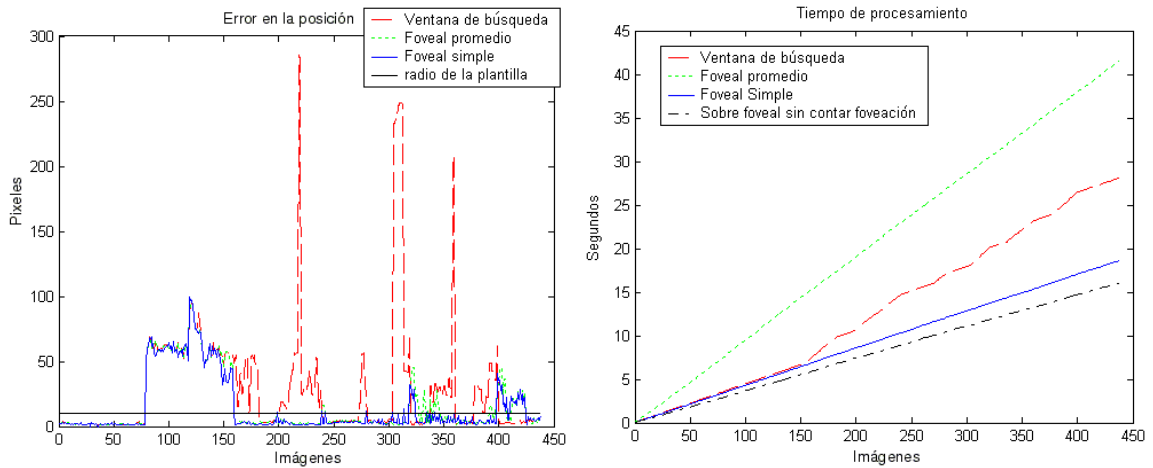


Figura 5.12: Resultados obtenidos para el seguimiento en la secuencia 2

Secuencia 2

Esta secuencia tiene un escenario similar al de la secuencia 1, pero con la diferencia de que la lancha viaja de derecha a izquierda y alrededor del centro de la escena, la lancha comienza a girar en círculos durante un cierto tiempo. Esto vuelve difícil la detección ya que el objeto cambia de forma continuamente. Posteriormente, el objeto interrumpe sus giros y viaja hacia la derecha del escenario. La figura 5.11, igual que la secuencia anterior, muestra imágenes de ejemplo de la detección en cada espacio correspondiente, cartesiano y foveal.

En la figura 5.12 se muestra el error en la posición del objeto para cada experimento. Después de la imagen 70 es cuando el objeto comienza a dar los giros y por lo tanto, debido a los cambios de forma, el algoritmo comienza a tener fallas tanto en cartesiano como en foveal al ser utilizado el módulo de detección de movimiento para identificar al objeto. Cuando la lancha comienza a bajar su velocidad en los giros, la detección mejora.

En esta secuencia un salto de 40 imágenes se introdujo por cada 80 imágenes. Esto es, después de la imagen 80 la siguiente imagen de la secuencia es la 120 y de ahí, después de la 200 se hizo un salto hasta la 240. Estos saltos producen la activación del módulo de detección de movimiento ya que la similaridad inmediatamente disminuyó debido a la desaparición repentina del objeto. Estos saltos se ven

reflejados en la gráfica del error en la posición del objeto en la figura 5.12. En ésta se muestran picos sobresalientes para las pruebas en el espacio cartesiano derivado de que el método tiende a equivocarse más a causa del movimiento adicional en el mar, lo cual provoca que la región de búsqueda se coloque equivocadamente sobre algunas regiones. El número de imágenes a saltar se eligió de tal manera que la posición siguiente del objeto fuese la primer coordenada donde el objeto cayese fuera de la ventana de búsqueda. Este experimento se realizó para demostrar el problema al que uno se enfrenta cuando la ventana de búsqueda debe ser ampliada para contener más información. La problemática como ya se ha dicho consiste en poder determinar qué tanto debe hacerse la ampliación y de qué manera, para que no produzca problemas en el proceso en tiempo real. Claramente el realizar la búsqueda sobre todo el campo de visión es una alternativa muy ineficiente, sobre todo considerando las dimensiones de la imagen cartesiana. Pero si realmente no se conoce la trayectoria del objeto previamente es difícil saber en que dirección buscar o que tanto ampliar la ventana. Se podría pensar en utilizar una ventana el doble de grande debido a que se sabe que el objeto cae apenas fuera de la ventana de búsqueda, pero el objetivo de este experimento es mostrar como afecta al tiempo acumulado (ver figura 5.12) el procesamiento de toda la imagen en el caso de que el sistema requiriera un análisis de al menos más de la mitad de la imagen (claramente las orillas no son de interés).

En el caso de las imágenes foveales, la detección es más eficiente ya que el movimiento es detectado en la zona periférica todavía con mayor presencia por encontrarse cerca de la región foveal (aunque no en ésta porque la región foveal tiene la misma área que la región de búsqueda por lo tanto el salto manda al objeto a la periferia). Del mismo modo, el muestreo decreciente en la periferia ayuda a filtrar el movimiento del mar que también introduce ruido al algoritmo de detección de movimiento, evitando así las falsas detecciones.

Secuencia 3

Esta secuencia presenta el problema de una oclusión que es mas grande que la ventana de búsqueda y por lo tanto, se origina el problema de la adaptación de la ventana de búsqueda. Para ésta secuencia, el escenario no causa mayor problema al algoritmo de detección que en el cartesiano, ya que realmente no hay mayor presencia

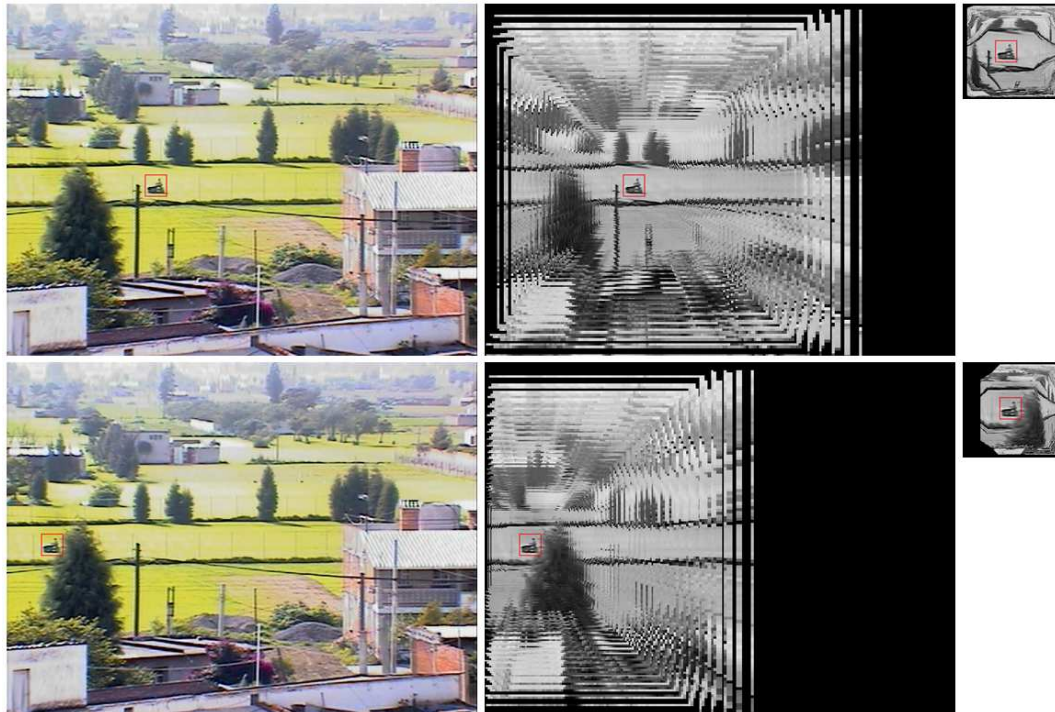


Figura 5.13: Secuencia 3

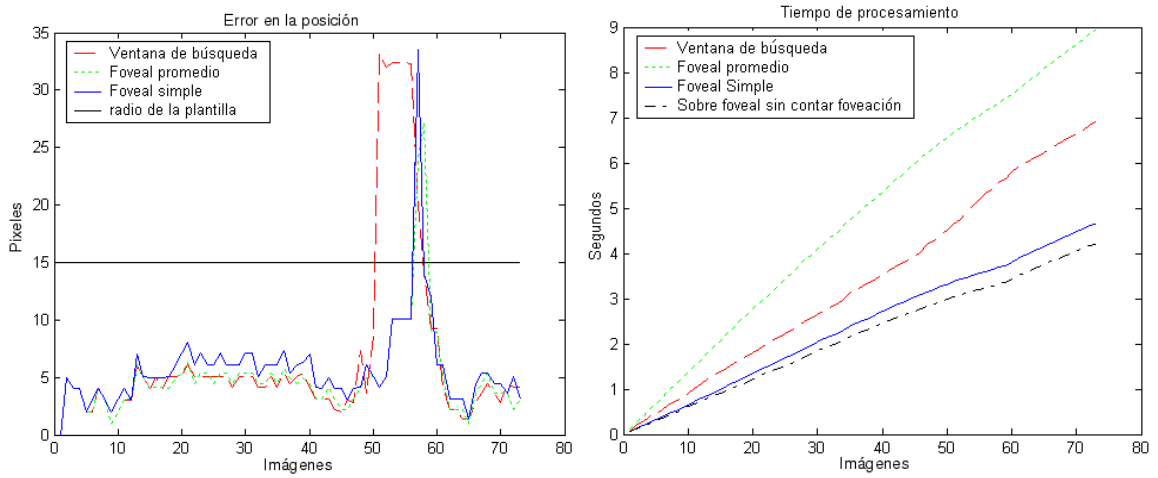


Figura 5.14: Resultados obtenidos para el seguimiento en la secuencia 3

de movimiento más que del objeto de interés. A pesar de que existen movimientos de árboles y sombras, pero éstos son filtrados por el mismo algoritmo de detección. De manera similar, para los experimentos con las imágenes foveales, el resultado en el seguimiento es el mismo pese a la oclusión.

Secuencia 4

En este escenario (ver figura 5.15) también se introdujo un salto en las secuencias. Cada 100 cuadros se omitieron 100 cuadros, ya que en esta ocasión el objeto tenía una dinámica de movimiento muy lenta relativamente. Lo importante de esta prueba es mostrar el problema de analizar el movimiento en todo el escenario cuando existe movimiento adicional al del objeto de interés. En el fondo de la imagen se observa un puente con automóviles que viajan en ambas direcciones produciendo trazas de movimiento.

Para el caso de los experimentos en el plano cartesiano, cuando los saltos se presentan, el algoritmo de detección de movimiento señala a estos objetivos como blancos probables debido a que su movimiento es mayor que el del mismo objeto de interés. Esto produce un falso en la detección, el cual se mantiene desde el primer salto hasta el final de la secuencia produciendo una detección completamente errónea.

Por lo contrario, los experimentos usando las imágenes foveales filtran nuevamente estos movimientos debido a que se encuentran en una región de la periferia con un nivel de detección realmente bajo. Esto hace que la detección de movimiento del barco conserve su jerarquía en la detección y permite realizar la recuperación y mantener el seguimiento a lo largo de la secuencia tal y como lo muestra la figura 5.16. La falsa detección en el plano cartesiano eleva los tiempos de procesamiento en el espacio cartesiano mientras que en el foveal el tiempo se mantiene bajo control.

Secuencia 5

Esta es una secuencia con fondo dinámico ya que la cámara se mueve gradualmente para mantener al objeto de interés dentro del campo de visión. El cambio de fondo es gradual y además el fondo es uniforme, se presenta sólo campo y cielo, o campo o cielo exclusivamente, ver figura 5.17.

El objeto de interés es un helicóptero (que en la vista tiene pequeñas dimensiones)



Figura 5.15: Secuencia 4

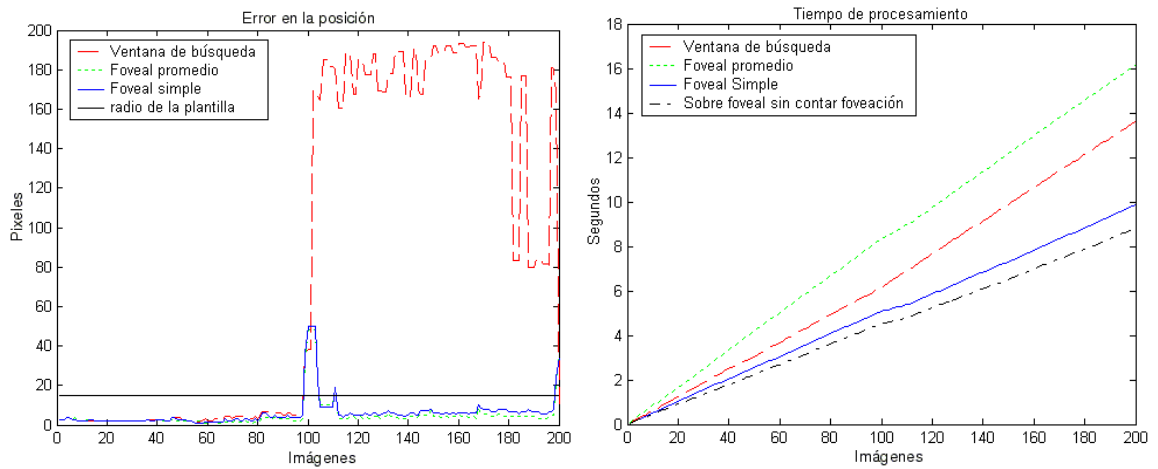


Figura 5.16: Resultados obtenidos para el seguimiento en la secuencia 4

que presenta movimientos no lineales con una dinámica y apariencia difícil de seguir. El helicóptero cambia su apariencia constantemente al dar giros laterales o rotando sobre si mismo. Esto implica que el módulo de detección mediante la similitud reportará valores muy bajos por lo que el módulo de detección de movimiento se utilizará con mayor frecuencia.

La detección en el espacio cartesiano reportó mayores detecciones erróneas que en el caso del foveal. El motivo nuevamente se deriva de que en el espacio foveal la región de mayor atención se encuentra situada sobre el objeto de interés, por lo que la detección de movimiento tiene mayor presencia, mientras que el resto del movimiento adicional producido por los cambios graduales en el fondo se filtra debido a la resolución decreciente. En el caso del espacio cartesiano, justo en la transición de fondos se obtiene una detección falsa de movimiento debido a la diferencia de imágenes aplicada. Este tipo de situaciones se refleja en los picos de la curva de error de posición en la detección.

De forma adicional este experimento muestra la ventaja de mantener una zona de atención con alta definición, pero a la vez conservar información del resto del campo de visión, ya que para ésta secuencia, el objeto cambia de tamaño continuamente acercándose y alejándose de la zona de atención. Cuando el objeto se acerca, el tamaño del objeto sobrepasa el área de búsqueda (ventana de búsqueda o región foveal). Para este tipo de situaciones mantener una buena heurística para saber que tanto agrandar la ventana de búsqueda y en que momento disminuirla resulta pesado en el espacio cartesiano. En contraste el espacio foveal permite superar éste problema, ya que el muestreo permite mantener la información con buena resolución para la parte del campo de visión cercana a la zona de atención usando una cantidad reducida de datos. Esto se demuestra en las curvas de error de posición en la figura 5.18.

Las curvas de tiempo acumulado de la figura 5.18 muestran nuevamente el problema de procesar todos los datos o la mayoría de los datos de la imagen en el caso del espacio cartesiano, mientras que en el espacio foveal el procesamiento resulta mucho menor.

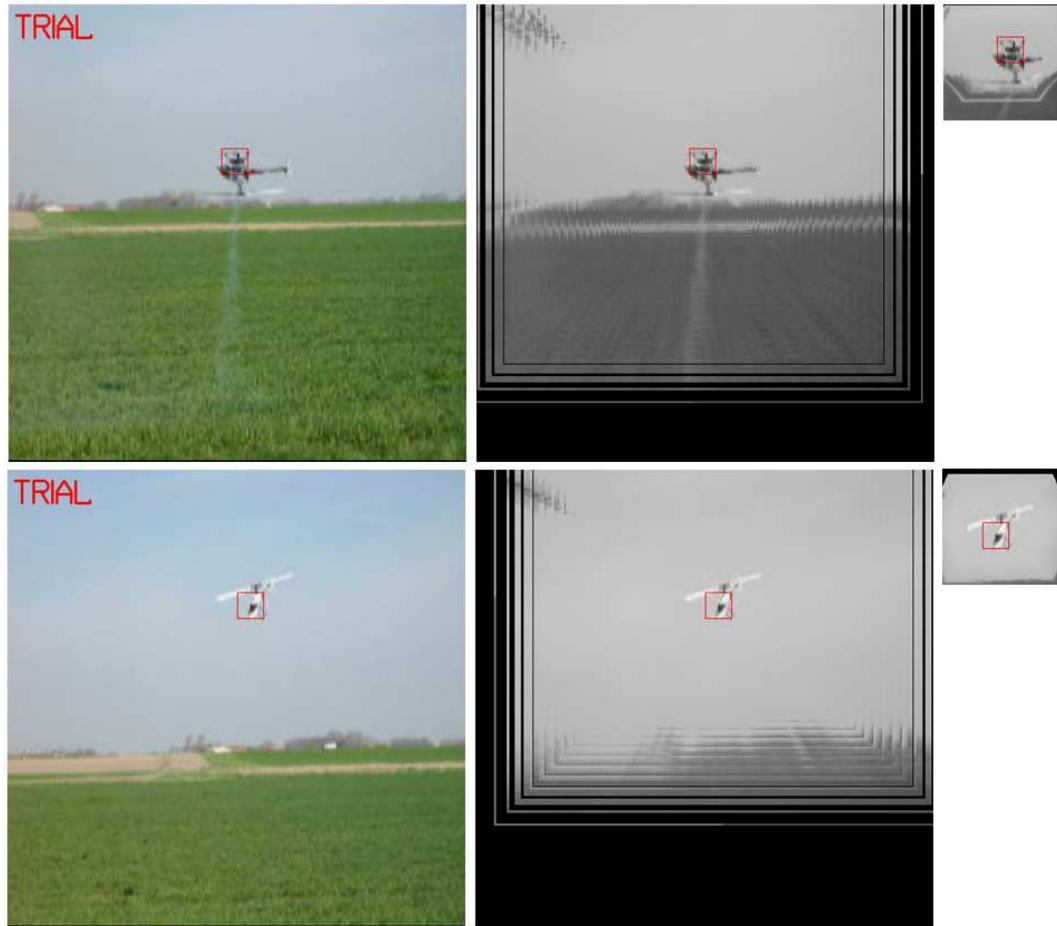


Figura 5.17: Secuencia 5

5.4.4. Discusión

Los experimentos presentados en esta sección muestran que las imágenes foveales propuestas permiten el uso de métodos lineales y basados en apariencia para la detección y el seguimiento. Es importante señalar que los experimentos se realizaron sobre el sistema operativo Windows XP y sobre una máquina con procesador AMD a 2.2 GHz y 1 Gb en memoria RAM.

Existen algoritmos más robustos para llevar a cabo el seguimiento pero el objetivo ha sido mostrar cómo un algoritmo sencillo tiene un buen desempeño que se refuerza por el espacio de trabajo de resolución decreciente proporcionado por las imágenes

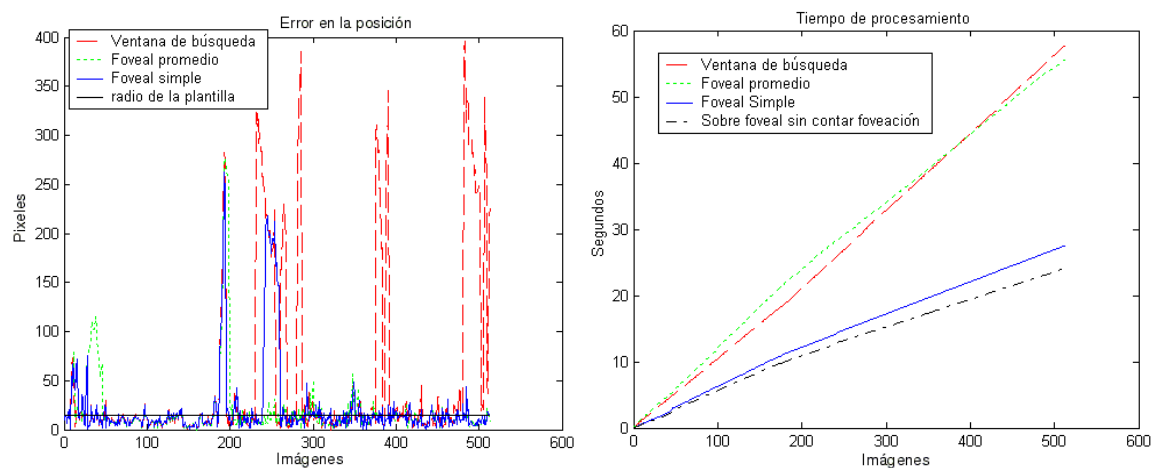


Figura 5.18: Resultados obtenidos para el seguimiento en la secuencia 5

foveales. Esto conduce a pensar en que el algoritmo de seguimiento utilizado se puede fortalecer con otras estrategias que sobre todo aprovechen la información jerárquicamente muestreada del campo de visión.

El uso del filtro Kalman en las imágenes foveales propuestas tiene el mismo desempeño que en las imágenes convencionales con la única diferencia de que los datos de detección deben ser sumados con el desplazamiento del centro foveal colocado sobre la coordenada (x_0, y_0) en la imagen cartesiana. Si se grafican las curvas de error de posición los resultados serán los mismos en razón de que el filtro no requerirá el uso de la región periférica y por lo tanto la detección sobre la región foveal se realizará tal como si se estuviera trabajando sobre la ventana de búsqueda.

Desafortunadamente, no siempre es posible complementar el algoritmo de seguimiento con el filtro Kalman. Nuevamente, el tiempo de respuesta y los recursos disponibles pueden ser factores que limiten su uso, adicional al problema de contar con el modelo dinámico del objeto de interés. Para muchas aplicaciones, dicho modelo es complejo de obtener como en el caso del helicóptero de la secuencia 5.

En las curvas de error de posición se observa que: el seguimiento efectuado sobre las imágenes foveales simples tiene un comportamiento muy similar al obtenido con las imágenes foveales promedio. Esto es lógico ya que las variaciones se dan cuando

entra a operar el módulo de detección de movimiento y, como se mostró en los experimentos de detección de movimiento, las imágenes foveales promedio tienen mejor desempeño. El problema es que, como se ha visto también en la gráficas de tiempo acumulado, si se utiliza la foveación promedio implementada por software, entonces el tiempo de procesamiento se elevará al procesar toda la imagen para obtener la imagen foveal. Por el contrario, si la aplicación permite una detección sin mucho rigor en la precisión y más que nada, de mantener una zona de atención sobre el objeto de interés, entonces las imágenes foveales simples resultan atractivas puesto que el tiempo de procesamiento es poco aún si la transformación se realiza por software. Independientemente de cual método de foveación se utilice, si se contará con algún hardware que permita realizar la foveación directamente de la señal de la cámara o de la tarjeta framegrabber, entonces el tiempo de procesamiento se reduciría considerablemente, como lo muestran las mismas gráficas de tiempo acumulado en la curva negra para cada experimento. Esto motiva al reto de no sólo proponer una geometría foveal o método de foveación que resuelva los problemas por los que la visión foveal ha dejado de ser utilizada, sino también que dicha geometría o método pueda ser llevado al hardware o bien incluso al área del cómputo reconfigurable.

Conclusiones

6.1. Conclusiones finales

Como se ha descrito en este trabajo la visión foveal fue introducida como un nuevo paradigma para resolver diversas tareas de visión por computadora sujetas a la fuerte restricción del tiempo real.

La idea se sustentó bajo la premisa de que los sistemas artificiales deben inspirarse en los sistemas naturales que trabajan con éxito como lo es el caso del sistema de visión natural.

La visión foveal referente del sistema de visión en los primates, ha sido modelada bajo diferentes propuestas con el fin de llevar el concepto natural a la implementación en algoritmos y estructuras de datos de la computadora. Sin embargo, en este trabajo de tesis se ha hablado de las severas limitantes que han surgido para su uso y que incluso han producido que el fuerte interés que surgió en un principio haya decaído los últimos años.

La falta de interés se puede ver reflejado en este último par de años en las conferencias de visión artificial internacionales más importantes donde, de todos los trabajos que se presentan, sólo uno o por mucho dos artículos tocan el tema de la visión foveal para resolver tareas como el seguimiento de objetos en tiempo real. La causa, como se ha explicado, se debe a que las diferentes propuestas implican algoritmos de mayor complejidad, lo que no es deseable en los sistemas de tiempo real que demandan algoritmos sencillos y eficientes con tiempos de respuesta óptimos.

En este trabajo se ha presentado una propuesta innovadora para llevar los conceptos de la visión foveal a la visión artificial, específicamente orientada a las tareas de

seguimiento. El método de foveación propuesto es simple y sencillo de implementar y se ha inspirado de las diferentes propuestas de foveación existentes. Se han mostrado experimentos que utilizan las imágenes foveales generadas con el método propuesto así como los resultados obtenidos, los cuales se consideran satisfactorios, y comparados contra los resultados que se obtienen utilizando imágenes de resolución uniforme. El primero de ellos es la reducción significativa de datos a procesar. El segundo y que se deriva de la reducción es que el tiempo de procesamiento se reduce a su vez.

Puede notarse que no se realizó alguna comparación en los experimentos de detección y seguimiento utilizando imágenes foveales generadas con otras propuestas. El motivo de esto es que se ha mostrado en el capítulo 2 las deficiencias y limitantes de cada una de esas propuestas (o al menos las más representativas), las cuales exponen que en suma - en lo que se refiere a la tarea de seguimiento - las aplicaciones son muy limitadas y no compiten realmente con las que utilizan imágenes cartesianas con algoritmos lineales y basados en apariencias.

A continuación se enlistan las aportaciones principales de este trabajo:

1. Se ha realizado un análisis del estado de la tecnología/ciencia concerniente a la visión foveal y su uso en la tarea de seguimiento de objetos, mostrando sus debilidades, logros y limitantes.
2. Se ha propuesto un nuevo método de foveación sencillo y práctico con características de fovea móvil, reajuste de muestreo y que además, permite una sencilla implementación no sólo por software, sino también en hardware (ver apéndice B).
3. Las imágenes foveales obtenidas permiten la utilización de métodos lineales y basados en apariencias, por lo que la detección y seguimiento de objetos se puede llevar a cabo con eficacia y con baja complejidad, situación contraria con otras representaciones.

Y las conclusiones que se han derivado de este trabajo:

1. Los resultados de este trabajo muestran que la propuesta planteada sí puede competir y que además puede ser de mucha utilidad para cierto tipo de aplica-

ciones cuya problemática y escenario de trabajo requieren de manera inherente un enfoque de visión foveal.

2. Se ha mostrado que los conceptos de visión foveal pueden ser ampliamente explotados para realizar tareas como la detección de movimiento y el seguimiento de objetos.
3. La resolución decreciente en combinación con algoritmos de detección y filtrado, permiten construir algoritmos simples y eficaces lo que también es ideal para las aplicaciones de tiempo real con recursos limitados.
4. Se han mostrado las ventajas de utilizar la visión foveal en virtud de sus propiedades de obtener información jerárquica de todo el escenario con un conjunto reducido de datos, lo que implica reducción de costos de procesamiento y tiempo.

6.2. Limitaciones

Se ha mostrado que la visión foveal combinada con ciertos algoritmos, permite resolver de manera sencilla aplicaciones como la detección y el seguimiento de un objeto. No obstante, existen otras aplicaciones para las cuales el muestreo decreciente no es adecuado. Como es el caso de la detección y seguimiento de múltiples objetos en un escenario, para lo cual es indispensable contar con la mayor información disponible de todo el escenario.

Por otra parte, el uso de la visión foveal debe corresponder únicamente a ciertas aplicaciones cuya problemática implican una región de atención a la par de contar con información del escenario para la toma de decisiones. De otro modo, las imágenes convencionales deberán seguir siendo utilizadas. Por esta razón, las imágenes foveales no deben ser vistas como un sustituto de las imágenes cartesianas. Se puede pensar incluso en sistemas que cambien de espacio de trabajo, invariante o variante, de acuerdo a las necesidades de la aplicación.

6.3. Perspectivas

Los experimentos realizados corresponden a secuencias de imágenes obtenidas con cámara fija, fondo estático o de cambio gradual. Esta fue una restricción establecida para este trabajo y como un trabajo a futuro se puede considerar el incluir algoritmos tales como compensación de movimiento o correspondencia de imágenes, con el fin de utilizar las imágenes foveales propuestas para el seguimiento y detección con cámara móvil o escenarios con fondo dinámico ampliando el conjunto de aplicaciones que se derivan de esta funcionalidad.

Las imágenes foveales propuestas pueden ser utilizadas en otras aplicaciones diferentes al seguimiento. Por ejemplo, se pueden utilizar para realizar la compresión de imágenes como lo propone [?]. La metodología puede consistir en convertir una imagen normal a una imagen foveal con el método propuesto y aplicar posteriormente un algoritmo de compresión de imágenes como el MPEG. El resultado sería una compresión que podría alcanzar una reducción mucho mayor. Una posible aplicación directa sería la transmisión de video en canales de comunicación con ancho de banda limitado.

Otra aplicación podría ser la utilización de las imágenes foveales para resolver el problema de la obtención del mapa de disparidad en imágenes estéreo. Debido a que las imágenes foveales propuestas preservan la relación espacial de los píxeles con una cantidad reducida de datos, es posible aplicar métodos basados en apariencia para encontrar la alineación entre las imágenes estéreo y construir así el mapa de disparidad en un tiempo menor a causa de la reducción en el volumen de datos. Esto será posible siempre y cuando el problema implique determinar la profundidad en 3D de una determinada zona del escenario como es propuesto en [?] por Bernardino.

Por último, las aplicaciones de seguimiento de un objeto pueden ser variadas en robótica, navegación, sistemas de vigilancia y sistemas de guiado entre otros. Realmente el trabajo a futuro en esta área sería el diseño e implementación de algoritmos más robustos y completos para realizar el seguimiento de un objeto de interés mediante el uso de las imágenes foveales propuestas.

Artículos publicados

Los siguientes artículos se generaron de los resultados obtenidos en este trabajo de tesis. Fueron presentados y publicados en conferencias internacionales los cuales se listan a continuación:

- *FPGA-based Pipeline Architecture to Transform Cartesian Images into Foveal Images by Using a new Foveation Approach.* **J. Martínez**, L. Altamirano. Proceedings of the International Conference on Reconfigurable Computing and FPGAs ReConFig 2006. Published by **IEEE**. San Luis Potosí, Mexico, September 2006.

Abstract: In Vision systems the image processing represents a bottle neck because the big amount of information that should be analyzed. Working with variant spaces over the visual field has been widely proposed as a way to reduce such information. Foveal vision is one of these proposals by providing a way to transform the visual field obtained with conventional cameras into a sampling with high resolution at the center and decreasing over the periphery such as in mammal vision systems. In this paper, an FPGA based architecture to transform conventional images into foveal images is presented. The hardware algorithm has been taken from new proposal to foveate images. Strategies as parallelism and pipeline are exploited to obtain a high performance and thus, with both of them, reduction in the visual field and the transformation in real time of the digital images into foveal images, a vision system can accelerate its performance and reaches real time restrictions.

- *A New Foveal Cartesian Geometry Approach used for Object Tracking.* **J. Martínez**, L. Altamirano. Proceedings of the 3rd IASTED Internacional Conference on Signal Processing, Pattern Recognition and Applications SPPRA 2006. Published by **ACTA Press**. Innsbruck, Austria. February 2006.

Abstract: Foveal vision has been used as a way for sampling and reducing the amount of data in Cartesian images for vision systems. For this sampling, there are different approaches as the Log Polar Transform, the Exponential Cartesian Geometry and the Foveal Wavelet Transform between others. In this paper a new approach to obtain the foveal sampling and its application to single object tracking is presented. The approach uses the log polar formulation for the sampling but preserves the Cartesian properties of the information in the original image. In this way, it is possible to overcome the problems of no-linearity of the Log Polar Geometry. Furthermore, it allows an easier object location as the hierarchical processing used in the Exponential Cartesian Geometries and it is easy to implement because it does not imply complex operations for the sampling of the images and the recovery of the original image through the foveated image, contrary to the Foveal Wavelet Transform. The proposed geometry has been tested in diverse images sequences where a single object is tracked successfully by appearance based methods which demonstrate the effectiveness of the proposal.

Implementaciones en hardware para la foveación

La incorporación de la visión foveal en la visión artificial ha sido motivada por las necesidades de procesamiento de imágenes en tiempo real [?]. El modo de incorporarla representa un reto y en suma implica los siguientes puntos:

1. Se debe permitir un muestreo variante con alta resolución al centro y decreciente en la periferia alrededor de dicha región.
2. El muestreo debe resultar en un conjunto de datos reducido y de fácil direccionamiento o referenciado.
3. Se debe contar con la capacidad de recuperar la posición espacial original de la información muestreada.
4. El muestreo debe ser simple y fácil de implementar tanto en software como en hardware.

En este apartado se describe la implementación del algoritmo hardware del método de foveación propuesto. La implementación en hardware de los métodos de foveación no es nuevo y de hecho como se explicó en el capítulo 2, es una necesidad para aquellas propuestas cuyo método de foveación resulta de procesamiento costoso si se implementa en software.

Si es el caso de que el método de foveación se implemente en software, el rendimiento de la detección en el caso de la tarea de seguimiento, puede verse afectado o bien, puede restar tiempo valioso para cualquier otra aplicación de tiempo real. Las

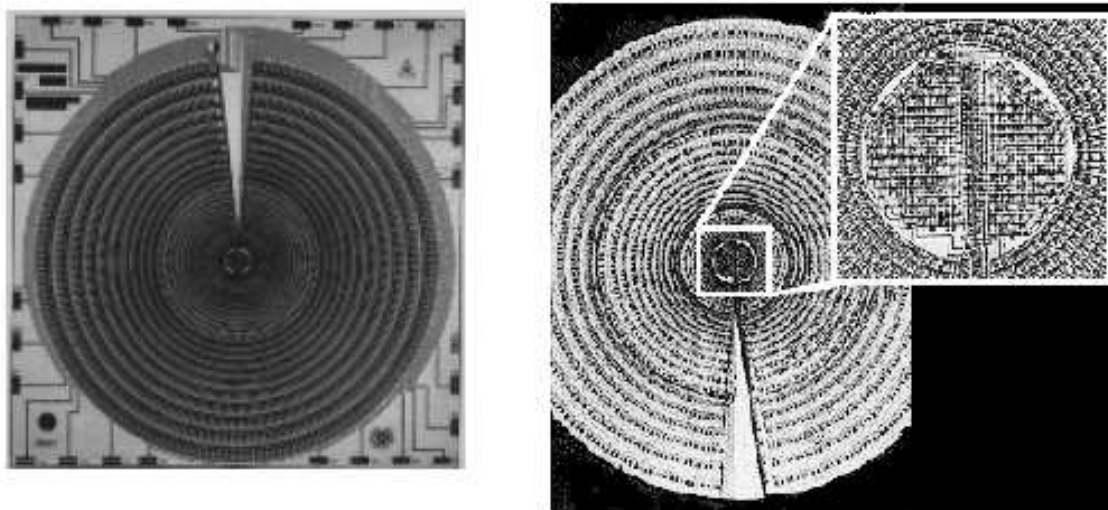


Figura B.1: Sensor CCD Retinal propuesto por Sandini a la izquierda y a la derecha un detalle de la región foveal

siguientes secciones mencionan brevemente estas propuestas en hardware y posteriormente, la implementación en hardware de la propuesta de foveación de este trabajo.

B.1. La transformada log polar

B.1.1. Sensores CCD log polares

El grupo de vision de Sandini es uno de los más representativos que ha trabajado con visión foveal. La Transformada Log Polar ha sido el objeto de estudio de todas sus publicaciones especializadas para el tema del seguimiento. El grupo de trabajo encabezado por Sandini desarrolló el primer sensor polar-logarítmico conocido como el *sensor CCD retinal* [?] y que emula a la retina del sistema de visión foveal natural. La figura B.1 muestra un ejemplo de la estructura CCD de este sensor. Este sensor posee una región interna, rectangular y con resolución variable formada por 30 círculos concéntricos constituidos cada uno de 64 fotodetectores. Este sensor tiene una tasa de captura de 50 cuadros por segundo, pero presenta algunos problemas como la discontinuidad entre la fovea y la periferia, y de que el área debe ser cortada en cada zona circular para permitir la conexión del reloj y las señales de control. En posteriores

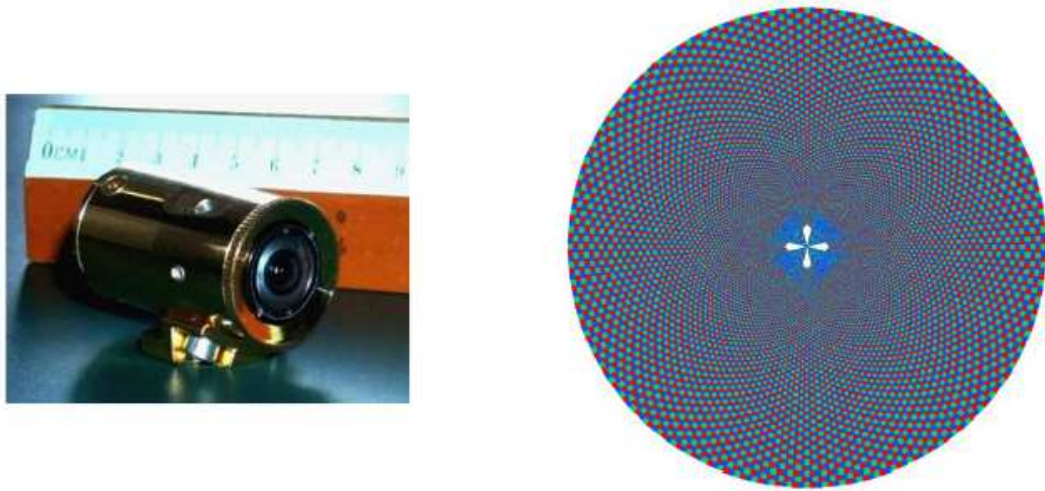


Figura B.2: Sensor CCD Retinal que captura color, la imagen de la izquierda muestra esquemáticamente la colocación de sensores para adquirir cada canal de color RGB

trabajos, el grupo presentó un modelo más avanzado de esta cámara log polar, pero ahora con la capacidad de capturar color. La figura B.2 muestra una configuración esquemática de los sensores colocados en la retina CCD para brindar la capacidad de captura del color.

Los consorcios IMEC e IBIDEM construyeron también un sensor polar logarítmico conocido como **FUGA18** [?]. Este sensor soluciona las deficiencias del primer sensor de Sandini con una solución de tecnología CMOS. Este sensor es de menor tamaño y menor consumo de energía, costos más bajos de fabricación y con la facilidad de integrar algoritmos de procesamiento sobre el mismo circuito integrado. Posee una región periférica formada por 56 círculos concéntricos, donde cada círculo tiene 128 fotoreceptores. También posee una región central foveal formada por 20 zonas circulares concéntricas de número variable de fotoreceptores: 1x4, 1x8, 2x16, 5x32 y 10x64 fotoreceptores. La figura B.3 muestra este sensor.

Por otro lado, un grupo de trabajo encabezado por Wodnicki en la universidad de McGill [?], construyó un sensor polar logarítmico que posee una región central uniforme formada por 40 columnas y 52 filas de fotoreceptores, así como una región periférica formada por 16 zonas circulares. Cada zona circular de esta periferia cuenta con 64 fotoreceptores. La principal ventaja de este sensor es que esta construido

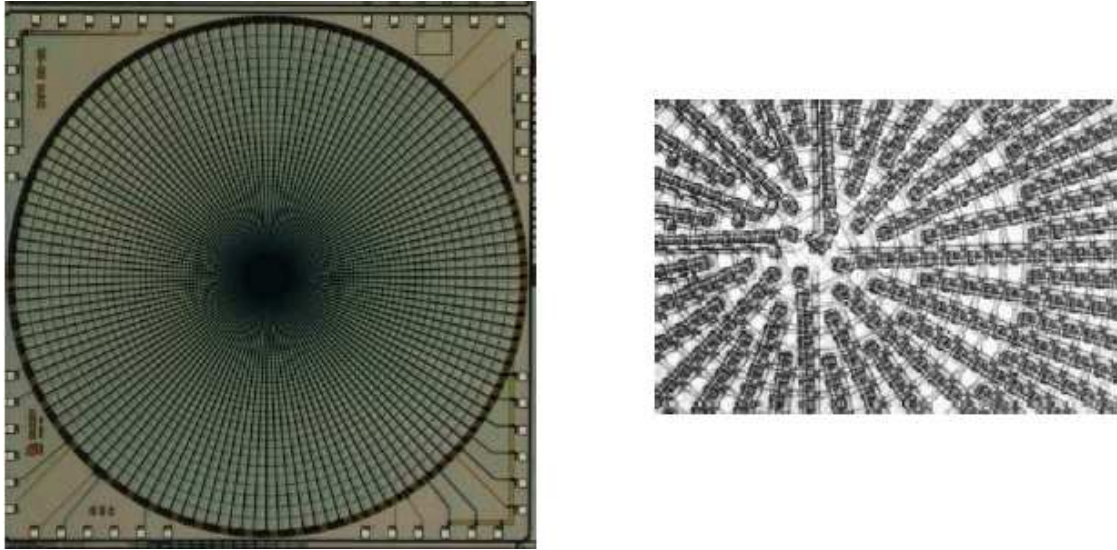


Figura B.3: Sensor CCD Retinal propuesto por Sandini a la izquierda y a la derecha un detalle de la región foveal

también con tecnología CMOS, pero nuevamente la discontinuidad entre los anillos de fotoreceptores. La figura B.4 muestra a este sensor que puede transmitir hasta 28 cuadros por segundo.

B.1.2. Implementación FPGA

La ventaja de contar con un sensor con tecnología CCD para el esquema log polar resulta atractivo para aplicaciones más directas de visión en tiempo real. Desafortunadamente, los costos de producción son elevados. Aunque la tecnología CMOS ha tendido a abarataarse en los últimos años, los sensores log polares no son comerciales aún por lo que no pueden ser producidos en serie para disminuir los costos de producción.

El cómputo reconfigurable ha brindado una solución atractiva y de bajo costo a los problemas de diseño y producción de hardware mediante el uso de las tarjetas FPGA (Field Programmable Gate Array) y las diversas herramientas de desarrollo ofrecidas por empresas como Xilinx.

En [?], Cobos y su equipo desarrollaron una arquitectura de hardware basada en FPGA para la Transformada Log Polar. Esta arquitectura es muy simple y consiste

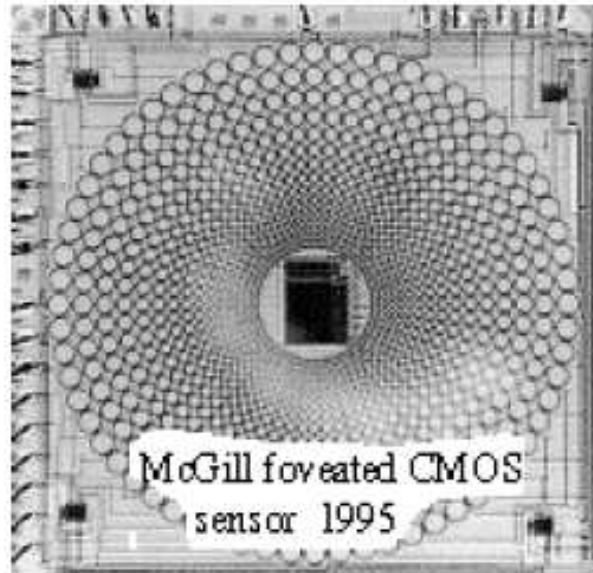


Figura B.4: Sensor polar logarítmico desarrollado por la universidad de McGill

en grabar en la memoria disponible en el FPGA las coordenadas cartesianas que son muestreadas en la relación uno a uno (foveación simple) por la Transformada Log Polar. Estas direcciones son traducidas a direcciones de memoria relativas al segmento de memoria en donde se almacena la imagen obtenida por la cámara a través del framegrabber. Posteriormente, una vez que la imagen está en memoria (memoria fuente), el FPGA lee en orden las direcciones de la memoria fuente y compara con las almacenadas en el FPGA para saber si el pixel corresponde a un pixel en la imagen log polar, si es así, el pixel es almacenado es transmitido a otra localidad de memoria (memoria destino), de otro modo no se hace nada. Para ambos casos, después del chequeo se procede con la siguiente dirección de memoria.

Esta implementación en realidad es muy sencilla y el inconveniente es que se encuentra limitada por la cantidad de memoria que pueda disponerse en el FPGA. La idea de esta implementación radica en el uso de tablas de precalculadas o Lookup Tables, que en este caso, equivale a grabar las coordenadas precalculadas por software y traducidas a direcciones de memoria de los pixeles que son muestreados por la TLP.

De cualquier modo, aún contando con las cámaras log polares y la implementación en FPGA, desafortunadamente la TLP cuenta por sí misma con puntos negativos que

se han descrito ya en la sección 2.3.3.

B.2. Geometría cartesiana exponencial

Las geometrías cartesianas exponenciales se presentaron como el mejor de los métodos de foveación que existen en la literatura. El gran inconveniente, aparte del procesamiento piramidal de las imágenes que puede demandar un poco más de análisis e implementación, es que la foveación piramidal obtenida con el polígono foveal [?] consume mucho tiempo de procesamiento.

Como se explicó en la sección 2.4.2, la pirámide se construye escalando diferentes áreas concéntricas de la imagen original. Estas reducciones se realizan en potencias de dos por lo que todo se reduce a un promediado de regiones de potencia de dos. Este promediado es costoso si se implementa en software, pero visto desde el punto de vista de hardware, el promediado y acceso en potencia de dos permite una manipulación natural de las direcciones de memoria gracias al sistema binario.

Esta ventaja ha permitido implementar algoritmos en hardware bajo un esquema similar al implementado para el log polar. Diversos trabajos han propuesto implementaciones en hardware basado en FPGA [?], implementaciones en arquitecturas MIMD [?] e incluso implementaciones en VLSI [?] entre las más representativas.

B.3. Implementación FPGA para el método de foveación propuesto

B.3.1. Estrategia de implementación

La implementación de los algoritmos en hardware se ha vuelto una realidad gracias a herramientas como *VHDL* (The Very High Speed Integrated Circuit Hardware Description Language) o el *ToolBox* para *MatLab System Generator* para el diseño de circuitos digitales en *simulink*, de la empresa *Xilinx* [?]. Este *ToolBox* se utilizó para diseñar y cosimular una arquitectura para el método de foveación simple para generar las imágenes foveales propuestas en este trabajo.

La idea se inspiró de la implementación FPGA de la TLP la cual propone seleccionar los píxeles de la memoria fuente de acuerdo a una dirección de memoria que se sabe, corresponde al muestreo foveal.

En síntesis, la estrategia consiste en generar o conocer de algún modo las direcciones de la memoria fuente que corresponden a los píxeles que deben ser muestreados y depositados en la memoria destino. Utilizar una tabla precalculada no resulta óptimo puesto que el FPGA debe realizar diversos accesos a memoria realizando en un porcentaje menor el procesamiento de datos.

La opción que se tomó fue la de programar el algoritmo en hardware equivalente al algoritmo en software del método de foveación propuesto. Para esto, se tienen dos alternativas: el método de foveación promedio y el método de foveación simple.

Debido a que el propósito de este trabajo es realizar el seguimiento con las imágenes foveales propuestas, y en virtud de los resultados obtenidos en los experimentos expuestos en el capítulo 5, el método de foveación simple fue elegido. Otra razón para la elección fue en razón del número de accesos a la memoria fuente que se origina al realizar el muestreo. En el caso de la TLP y las GCE, la memoria fuente tiene que ser recorrida completamente por lo que el número de accesos totales será igual al número de bytes que ocupe la imagen cartesiana en la memoria.

La ventaja de utilizar el método simple es que el muestreo implica una relación de uno a uno, por lo que no es necesario revisar todos los píxeles en la imagen para asignar el píxel correspondiente a la coordenada foveal. Pero aún cuando se programe el método de foveación simple, no se evita el chequeo completo de la memoria fuente para discriminar los píxeles que deben ser muestreados. El procedimiento bajo esta propuesta sería tomar una dirección de la memoria fuente, transformarla a su equivalente en foveal y asignarla la dirección a la dirección de la memoria destino.

Algo más eficiente puede hacerse si se toma el problema al revés por así decirlo esto es, generar revisar las direcciones de la memoria destino y transformarlas a su coordenada cartesiana con su correspondiente dirección en la memoria fuente. De este modo, el mapeo uno a uno se hace implícito y que en realidad corresponde al algoritmo de foveación inversa explicado en la sección 4.2.2.

La estrategia a seguir consiste en recorrer la memoria destino mediante la generación en orden de sus direcciones de memoria. La generación es sencilla ya que se puede iniciar un contador en cero hasta el número de elementos necesarios en la

memoria destino. Se empieza desde cero para poder producir sin mayor problema las coordenadas foveales (f_x, f_y) correspondientes. Al final del mapeo, a la coordenada (x, y) se le sumará la posición central de la fovea (x_0, y_0) en la imagen cartesianas de acuerdo a lo establecido con las propiedades de fovea móvil y así, la coordenada final es convertida a la dirección en la memoria origen correspondiente.

Para generar el recorrido de la memoria se diseñó un módulo que genera las coordenadas (x, y) en orden mediante un contador que comienza desde 0 y termina hasta el número de píxeles en la imagen foveal. Estas coordenadas viajan a través de un camino de datos, pasando por diversos componentes que modifican su valor mediante las operaciones del algoritmo de foveación inversa hasta que son convertidas nuevamente en una dirección de memoria para indicar el dato a muestrear de la memoria origen y colocarlo en la memoria destino con la dirección del componente que generó las coordenadas (x, y) iniciales (ver diagrama de la figura B.5).

El siguiente pseudo código muestra el algoritmo desglosado de foveación inversa y explicado en la sección 4.2.2. Se han enumerado las líneas para identificar de manera individual las operaciones requeridas con el fin de mostrar como se llevó a cabo la traducción a algoritmo de hardware.

Algoritmo de Foveación Inversa

1. $x_f = x_f - x_c$
2. $y_f = y_f - y_c$
3. $\rho = \text{máx}(|x_f|, |y_f|)$
4. Si $\rho \leq \rho_0$ entonces
5. $\tau = \rho - \rho_0$
6. $x = x_f + x_0$
7. $y = y_f + y_0$
8. *else*
9. $x = x_f \Delta \rho_\tau + x_0$

10. $y = y_f \Delta \rho_\tau + y_0$

11. *return*(x, y)

B.3.2. Uso de tablas precalculadas

El primer punto a observar en el algoritmo, es que el factor $\Delta \rho_\tau$ es un arreglo con cantidades decimales. Este arreglo puede ser precalculado, siguiendo las fórmulas de la sección 4.2.2, en función de todos los posibles valores de τ que van de $1 \dots RN_r$, ya que estos factores se mantendrán constantes hasta que se decida cambiar la configuración de la fovea en el número de anillos y el radio del área foveal. Por este motivo se han saltado los pasos 3 y 4 de la descripción en la sección 4.2.2

Tomando la idea de la tabla de direcciones precalculadas del algoritmo FPGA para la TLP [?], el vector $\Delta \rho_\tau$ puede ser calculado por software previamente y posteriormente ser almacenado en una memoria RAM en el FPGA. Esto resulta más simple que realizar el cálculo dentro del mismo FPGA ya que se deben realizar diversas operaciones complejas para ser implementadas en el hardware (logaritmos, potencias, divisiones, multiplicaciones) y además, almacenando en memoria el espacio que se ocupa es mucho menor.

Para esto se estableció que el número máximo de anillos para la foveación es de 128 y por lo tanto $\Delta \rho_\tau$ tendrá a lo más 128 elementos, lo que implica un espacio de memoria de 128 localidades. En cuanto a las localidades, para el manejo de los números decimales se utilizó la representación de punto fijo. Con esto, se estableció una longitud de palabra de 16 bits con 8 bits para la parte entera y 8 bits para la parte decimal. Los experimentos de foveación demostraron que esta representación fue suficiente para llevar a cabo la foveación.

B.3.3. Traducción de operaciones en componentes de hardware

Dejando fuera el cálculo de $\Delta \rho_\tau$ y las multiplicaciones por este factor en las líneas 9 y 10, las operaciones restantes son sumas, comparaciones, restas y valores absolutos que son sencillos de traducir en componentes de hardware. También se puede observar

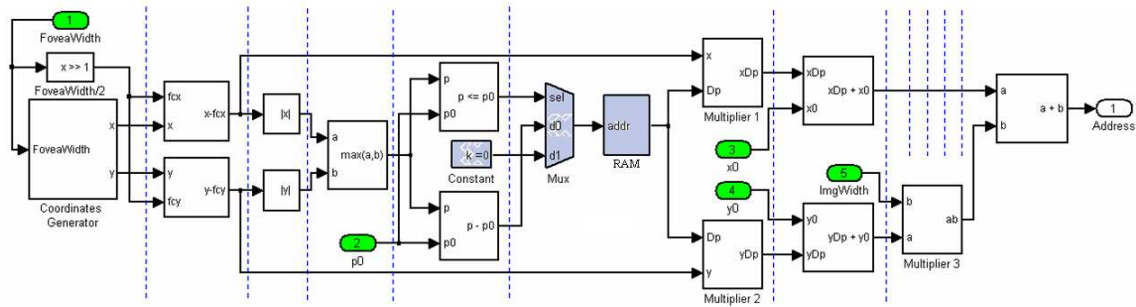


Figura B.5: Diagrama de componentes digitales de la arquitectura FPGA del algoritmo de foveación inversa

que las variables x y y no dependen una de otra por lo que su cálculo puede realizarse en paralelo.

La figura B.5 muestra un diagrama a bloques de los componentes digitales que fueron diseñados utilizando la herramienta Simulink de Matlab y con componente para diseño lógico digital de Xilinx.

La líneas azules punteadas en el diagrama indican que en esa sección del camino de datos se han colocado registros para producir el efecto de ejecución en pipeline. El objetivo es disminuir la longitud del ciclo de reloj ya que de otro modo la arquitectura presentaría una ejecución muy lenta.

En el diagrama se observa la memoria RAM en la que se graban los valores de $\Delta\rho_\tau$ y que son direccionados mediante la salida del multiplexor elegida por el valor en 1 si la $\rho \leq \rho_0$ y 0 en caso contrario. Cuando no es menor, el valor de salida en el multiplexor es un '0' para acceder a la dirección cero de la memoria RAM. En esta dirección se almacena un 1. A partir de la dirección 1 hasta la 127, se almacenan los diferentes valores de $\Delta\rho_\tau$.

B.3.4. Multiplicador con estrategia pipeline

Para la multiplicación, Xilinx pone a disposición un componente que multiplica dos números binarios. Mediante una configuración adecuada, el componente obtendrá un número binario con la longitud de palabra y la precisión de punto fijo correspondientes.

El problema con este componente es que tiene una latencia de 4 ciclos para generar

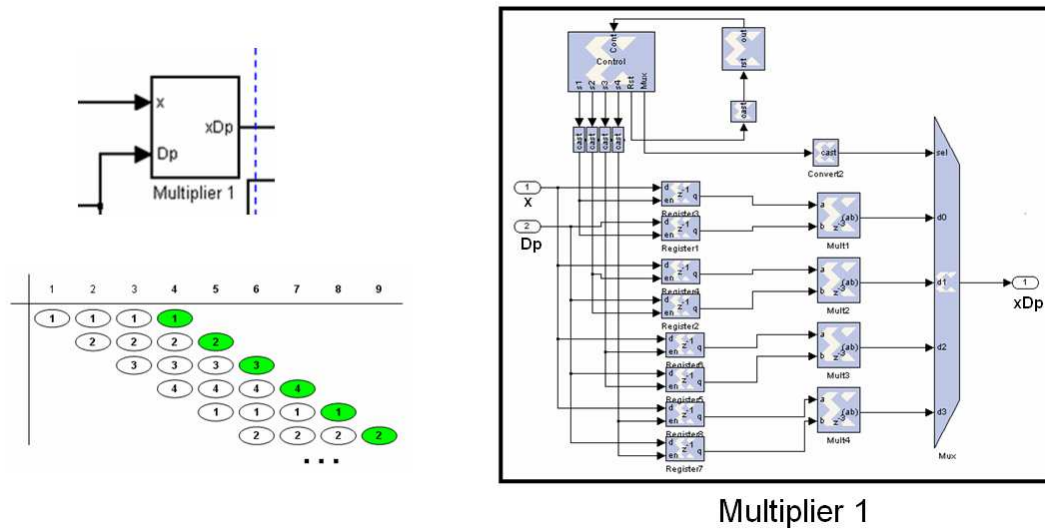


Figura B.6: Esquema del multiplicador construido de un arreglo de multiplicadores para producir resultados en pipeline

la salida de la multiplicación por lo tanto, se debe idear una estrategia para detener el flujo del camino de datos hasta que el multiplicador este disponible para procesar nuevos. Esto produce un inconveniente ya que hay 5 etapas antes de que los datos lleguen al multiplicador y por lo tanto, existirá una latencia de 5 ciclos para la llegada y posteriormente otros 4 para la salida del multiplicador por lo que, los datos de entrada deben ser encolados para evitar detener el flujo del camino de datos, en caso contrario, el pipeline pierde sentido.

La figura B.6 muestra un componente que se diseño para resolver el problema de la latencia del multiplicador. La idea consiste en construir un arreglo de multiplicadores (multiplicador componente de Xilinx) con dos registros conectados a sus respectivas entradas. Los datos de entrada principales del componente (x y Dp en la figura) se conectan por parejas a los registros en cada componente y en paralelo. Este componente tiene un control que habilita la escritura a la pareja de registros correspondientes mediante un contador. El contador se incrementa con cada ciclo de reloj. De este modo, al primer ciclo, los registros del multiplicador uno se habilitaran para grabar la entrada de datos principal, mientras todos permanecen bloqueados. En el siguiente ciclo, el segundo par de registros es habilitado para realizar la recepción y envío de datos al segundo multiplicador mientras todos los demás registros per-

manecen bloqueados. Mientras esto sucede, el primer multiplicador llevará dos ciclos de procesamiento en la multiplicación. Al tercer ciclo, el control habilita al tercer par de registros para grabar la salida mientras bloquea todos los demás registros. El primer multiplicador llevará 3 ciclos de operación, el segundo llevará dos y el tercero iniciará con el primer ciclo. Al cuarto ciclo, cuando se habilite el cuarto par de registros y se deshabilite el resto, el primer multiplicador producirá el resultado del producto y el contador reiniciará a 0 para habilitar el primer par de registros en el siguiente ciclo ya que el primer multiplicador quedará disponible.

La descripción anterior puede visualizarse en la figura B.6 en la tabla que tiene círculos dibujados. Esta tabla muestra en la cabecera los ciclos de reloj mientras en las filas se muestran los ciclos que consume cada multiplicador así como los ciclos en los que inicia su actividad. En los círculos rellenos se ha señalado el momento en el que producen la salida cada uno de los multiplicadores. Esta estrategia evita el encolamiento y brinda una solución en pipeline con una latencia de 4 ciclos para comenzar a producir resultados en serie. Por lo tanto, para obtener los primeros resultados hasta la etapa de multiplicación pasarán 9 ciclos para comenzar a generar datos de manera consecutiva.

A la salida de la multiplicación le siguen otras dos etapas que consisten en sumar el centro de la región foveal sobre la imagen cartesiana en las coordenadas ya que hasta ahora, las coordenadas obtenidas se encuentran relativas al centro $(0,0)$ y hay que trasladarlas a la coordenada (x_0, y_0) .

La última etapa consiste en convertir las coordenadas cartesianas en una dirección de memoria y para ello se aplica un polinomio de direccionamiento clásico. Nuevamente, se utiliza un componente multiplicador como el de la figura B.6. En este componente se multiplica el ancho por el número de filas y su salida será sumada con el número de columnas. A causa de la latencia de 4 ciclos, para que el dato no llegue inmediatamente al componente de suma se colocan 4 registros intermedios como lo muestra la figura B.5. Los registros se representan por las cuatro líneas punteadas sobre la conexión entre el componente que suma $xDp + x_0$ y el último componente del camino de datos. De este modo, cuando el multiplicador termine su procesamiento, al mismo tiempo llegarán al último componente el valor de la coordenada en x a sumar con el resultado de la multiplicación. El valor final de esta operación de suma corresponderá a la dirección en la memoria fuente que debe ser muestreado para colo-

carlo en la memoria destino. En total, para obtener este resultado final, la latencia de respuesta será de 13 ciclos a partir de los cuales se generará de manera continua las direcciones de memoria a muestrear.

B.4. Resultados obtenidos

La arquitectura recibe como entrada 4 parámetros de configuración que se describen en la tabla B.1

Tabla B.1: Parámetros de configuración para el método de foveación

Parámetro	Significado
ρ_0	Radio de la región foveal.
x_0, y_0	Coordenadas cartesianas donde se colocará el centro de la fovea
FoveaA	Ancho en pixeles de la imagen foveal.
ImagenA	Ancho en pixeles de la imagen cartesiana.

La arquitectura fue diseñada para procesar imágenes digitales de hasta 640x480 y 512x512 pixeles. Además, con la variación de los parámetros descritos en la tabla B.1, se pueden generar imágenes foveales de hasta 255x255 pixeles.

La frecuencia máxima es de **74.17 Mhz** lo que implica que, para almacenar y procesar una imagen de 640x480 pixeles con una región foveal de hasta 255x255 pixeles, se requiere un total de 640x480 ciclos para almacenamiento de la imagen digital en la memoria fuente, más 255x255 ciclos de la generación de las coordenadas de muestreo en la foveación inversa, más 13 ciclos de latencia de la arquitectura. Esto da un total de 372238 ciclos y bajo la frecuencia máxima de trabajo se obtiene un total de **5 ms** para capturar y fovear una imagen digital, lo suficiente para trabajar con más de 24 cuadros por segundo, requerimiento mínimo para trabajar con tiempo real en diversas tareas de visión. Más aún, este es un resultado interesante ya que el tiempo de foveación en software asciende a **18 ms** en promedio por imagen. Esto significa una reducción substancial en el tiempo de foveación lo que puede ser aprovechado para trabajar con los algoritmos de seguimiento y detección como si se estuviera trabajando con la captura directa de la cámara. Otra ventaja de esta arquitectura es el espacio utilizado en memoria del FPGA que es de **256 bytes** (128 localidades de

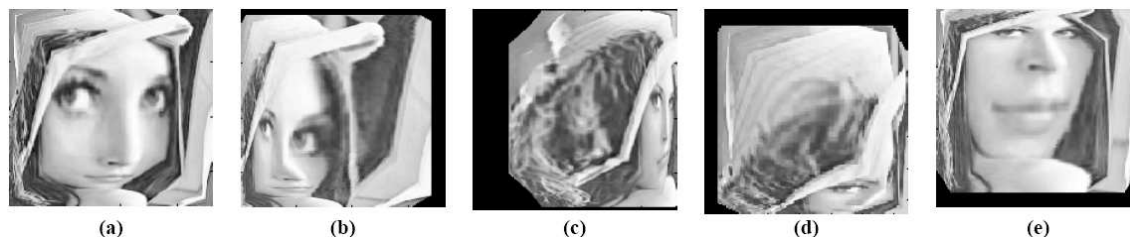


Figura B.7: Ejemplos de imágenes foveales obtenidas con la arquitectura propuesta

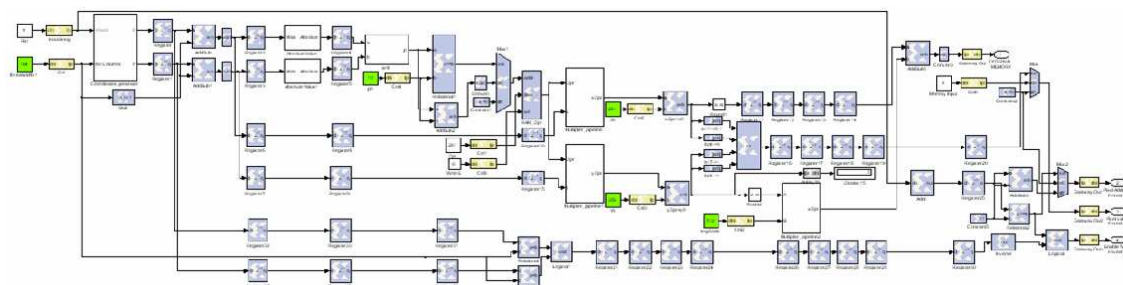


Figura B.8: Arquitectura diseñada con las librerías de Xilinx para MatLab

memoria por 16 bits ancho de la palabra).

Tabla B.2: Configuración para los experimentos realizados y similitudes obtenidas

Imagen	(x_0, y_0)	ρ_0	N_r	$ \Delta\rho_r $	Dim. Fovea	Similaridad
a	256,256	30	60	60	181x181	0.9998
b	336,217	30	60	60	181x181	0.9996
c	83,261	30	60	60	181x181	0.9988
d	167,145	15	50	49	129x129	0.9997
e	261,356	10	80	72	165x165	0.9995

La figura B.7 muestra algunos ejemplos de foveación realizados con la arquitectura FPGA propuesta. Las imágenes foveales se generaron variando los parámetros de acuerdo a los valores que se muestran en la tabla B.2, la imagen cartesiana original es de 512x512 píxeles. Las imágenes obtenidas con la arquitectura fueron comparadas por las obtenidas por software utilizando el algoritmo de foveación simple. Para la comparación, se utilizó la medida de similitud enunciada en la ecuación 3.3.2. Los

resultados muestran la alta similitud entre las imágenes obtenidas por software y las obtenidas por la arquitectura. Estos resultados muestran la eficacia de la arquitectura.

Por último, la figura B.8 muestra el diseño con componentes de Xilinx de la arquitectura presentada en este apéndice que fue co-simulada en una tarjeta FPGA de la familia Spartan 3. La tabla B.3 muestra algunos datos obtenidos del reporte "place and route".

Tabla B.3: Datos obtenidos por el reporte Place and Route

Device	Resources	Usage
Xc3s200-5ft256	Usage/Total	(%)
ExternalIOBS	116/173	67%
RAMB16s	1/12	8%
Slices	1714/1920	89%
BUFGMuxs	1/8	12%
Max. path delay	Max. Clock Frequency	
13.48 ns	74.17 Mhz	

B.5. Discusión

En suma, en este apéndice se ha presentado el trabajo desarrollado en esta tesis con respecto a la implementación del método de foveación propuesto en hardware. En este trabajo de tesis se ha discutido la importancia de contar con esta alternativa de implementación más que sólo contar con la de software. Los resultados obtenidos evidencian que es posible implementar esta geometría de modo sencillo, práctico y a bajo costo.

Como trabajo a futuro se propone complementar este método de foveación con otros módulos de visión como detectores de movimiento, medidas de similitud o extractores de bordes con el objetivo de diseñar un chip o tarjeta embebida que contenga un modulo completo de seguimiento o detección complementado con visión foveal.

