



# Video Tracking

**Laura Reguera García**

5º Ingeniería Industrial – VISIÓN COMPUTACIONAL

Memoria teórica

Abril 2010

Las técnicas de visión por computador tienen como fin último extraer propiedades del mundo a partir de un conjunto de imágenes. Una de las aplicaciones más importantes de la visión artificial es aquella relacionada con el seguimiento de objetos móviles. Un sistema de visión es capaz de obtener información de la realidad sobre colores, brillo, formas...Partiendo de la suposición de que el movimiento de una determinada escena es continuo, es posible hacer predicciones sobre el movimiento de los puntos de la imagen, en un instante determinado.

El análisis del movimiento está relacionado, entre otros, con aplicaciones en tiempo real como la navegación y seguimiento y con la obtención de información sobre los objetos estáticos y en movimiento en una escena. El guiado de un vehículo autónomo, la evaluación automática de la calidad de una pieza de cerámica o la inmersión automática de un personaje gráfico en una película, son algunos ejemplos de aplicaciones actuales de la visión por computador.

El análisis del movimiento es también fundamental en problemas como la restauración de secuencias de imágenes, así como su compresión y la obtención de imágenes y secuencias de imágenes de alta resolución a partir de secuencias de baja resolución.

El seguimiento de objetos en secuencias de imágenes se puede estudiar mediante varios métodos como pueden ser: el Filtro de Kalman o el Filtro de Partículas entre otros. El Filtro de Kalman define un conjunto de ecuaciones matemáticas que proveen una solución recursiva computacionalmente eficiente del método de mínimos cuadrados. El Filtro de Partículas es un método secuencial de Monte Carlo aplicable a cualquier transición de estados o modelo de medida.

# ÍNDICE GENERAL

<b>1.</b>	<b>INTRODUCCIÓN.....</b>	<b>1</b>
1.1	SISTEMA DE VISIÓN ARTIFICIAL.....	1
	▪ <i>Imagen digital.....</i>	<i>1</i>
	▪ <i>Elementos de un sistema de visión artificial.....</i>	<i>2</i>
	▪ <i>Etapas de un proceso de visión computacional.....</i>	<i>3</i>
<b>2.</b>	<b>APROXIMACIÓN AL PROBLEMA .....</b>	<b>4</b>
2.1	SISTEMA DE VISIÓN DINÁMICA.....	4
	▪ <i>Subsistemas .....</i>	<i>4</i>
	▪ <i>Variables en un sistema de visión dinámica.....</i>	<i>5</i>
2.2	ANÁLISIS DE ESCENAS DINÁMICAS .....	5
<b>3.</b>	<b>VISIÓN DINÁMICA .....</b>	<b>6</b>
3.1	MÉTODOS DE DETECCIÓN DEL MOVIMIENTO.....	6
3.2	DETECCIÓN DE CAMBIOS.....	7
	▪ <i>Sustracción de fondo.....</i>	<i>7</i>
	▪ <i>Imagen diferencia y diferencias acumuladas .....</i>	<i>7</i>
	▪ <i>Ajuste de bloques o block matching.....</i>	<i>8</i>
3.3	SEGMENTACIÓN .....	9
3.4	CORRESPONDENCIA .....	10
3.5	FLUJO ÓPTICO .....	10
	▪ <i>Problemas: iluminación y apertura.....</i>	<i>10</i>
	▪ <i>Ecuación de restricción del flujo óptico (ERFO).....</i>	<i>11</i>
	▪ <i>Técnicas basadas en gradiente.....</i>	<i>11</i>
	▪ <i>Técnicas basadas en correlación.....</i>	<i>11</i>
	▪ <i>Técnicas basadas en frecuencia.....</i>	<i>11</i>
3.6	SEGUIMIENTO .....	12
	▪ <i>Coherencia del camino .....</i>	<i>12</i>
	▪ <i>Filtro Kalman.....</i>	<i>12</i>
	▪ <i>Filtro de partículas .....</i>	<i>13</i>
<b>4.</b>	<b>HERRAMIENTAS Y LIBRERÍAS SOFTWARE.....</b>	<b>15</b>
4.1	PLATAFORMAS DE DESARROLLO DE CONTENIDOS .....	15
4.2	SOFTWARE DE INTEGRACIÓN Y TRACKING DE CÁMARA.....	15
4.3	LIBRERÍAS DE VISIÓN ARTIFICIAL Y TRACKING DE PUNTOS.....	16
<b>5.</b>	<b>APLICACIONES.....</b>	<b>17</b>

6.	<b>CONCLUSIONES</b> .....	<b>18</b>
7.	<b>RECURSOS DE INTERÉS</b> .....	<b>19</b>
	▪ <i>Artículos</i> .....	<i>19</i>
	▪ <i>Videos</i> .....	<i>19</i>
	▪ <i>Códigos</i> .....	<i>19</i>
	▪ <i>Noticias relacionadas</i> .....	<i>19</i>
8.	<b>BIBLIOGRAFÍA</b> .....	<b>20</b>

## ÍNDICE DE FIGURAS Y TABLAS

FIGURA 1	EJEMPLO SECUENCIA DE IMÁGENES .....	1
FIGURA 2	SISTEMA DE VISIÓN DINÁMICA BASADO EN ARQUITECTURA PC .....	2
FIGURA 3	ETAPAS DE UN PROCESO DE VISIÓN COMPUTACIONAL .....	3
FIGURA 5	TABLA DE VARIABLES DE UN SISTEMA DE VISIÓN DINÁMICA .....	5
FIGURA 6	CLASIFICACIÓN DE LOS MÉTODOS DE ESTIMACIÓN DE MOVIMIENTO .....	6
FIGURA 7	ESQUEMA BÁSICO DEL ALGORITMO BLOCK-MATCHING .....	8
FIGURA 8	EJEMPLO SEGMENTACIÓN ( <a href="http://blogs.mathworks.com/steve/2006/06/02/cell-segmentation/">HTTP://BLOGS.MATHWORKS.COM/STEVE/2006/06/02/CELL-SEGMENTATION/</a> ).....	9
FIGURA 9	DESPLAZAMIENTO EN LA FASE EN UNA SEÑAL DE VÍDEO .....	12
FIGURA 10	FILTRO KALMAN .....	13
FIGURA 11	EJEMPLO DE MUESTREO Y PESOS OBTENIDOS AL APLICAR EL FILTRO DE PARTÍCULAS....	14
FIGURA 12	RESULTADO DE UNA PRUEBA REALIZADA CON UNA PELOTA .....	14
FIGURA 13	INTERFAZ DE BLENDER .....	15
FIGURA 14	CAMPOS DE APLICACIÓN DE LA VISIÓN DINÁMICA.....	17

## 1. INTRODUCCIÓN

La visión artificial o visión computacional es una disciplina que tiene como finalidad la extracción automática de información del mundo real a partir de imágenes, utilizando como herramienta un ordenador. Abarca muchos y muy diversos usos, como pueden ser la detección y reconocimiento de objetos, evaluación de resultados, mapeo de imágenes...etc.

La aplicación de la visión artificial que se presenta en este trabajo es el seguimiento de objetos de una escena dinámica. Se trata de identificar de manera continua y a partir de una secuencia de imágenes o fotogramas relacionados entre sí, la posición de un objeto.



Figura 1 Ejemplo secuencia de imágenes

Para procesar la información dinámica se puede proceder de tres maneras diferentes. Se pueden aplicar varias técnicas estáticas a cada imagen por separado, analizar la secuencia de imágenes de forma continua o aplicar ambas técnicas en conjunto.

### 1.1 Sistema de visión artificial

Desde un punto de vista técnico, un sistema de visión computacional o artificial es un conjunto de elementos que permiten obtener imágenes del entorno, procesarlas y tomar ciertas decisiones basadas en la evaluación de las imágenes adquiridas. Es decir, un sistema autónomo que sea capaz de realizar alguna de las tareas de visión humana.

- ***Imagen digital***

La imagen digital es la proyección en dos dimensiones de una escena tridimensional. La reducción de una dimensión supone la pérdida de gran cantidad de información lo cual dificulta el proceso de visión. Este es el principal inconveniente que nos encontramos al intentar reproducir el resultado de la visión humana a través de un sistema de percepción electrónico.

La representación de la información de una imagen se puede realizar de varias formas pero independientemente de su formato, se representa por una matriz de puntos (denominados *pixeles*).

▪ **Elementos de un sistema de visión artificial**

Los componentes principales de un sistema de visión artificial son un sensor de imagen y un digitalizador, lo que viene siendo en términos coloquiales el ojo y el cerebro del sistema.

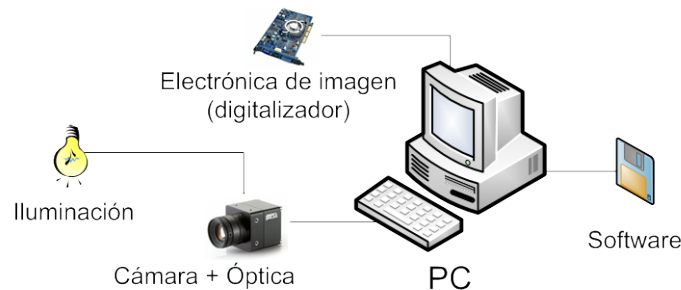


Figura 2 Sistema de visión dinámica basado en arquitectura PC

En una arquitectura básica distinguimos dos elementos: el sistema de formación de imágenes o software, y el hardware o sistema de procesamiento de éstas. El hardware está formado por el sistema de iluminación, captación de imagen y adquisición de la señal. Es el encargado de recoger las características del objeto en estudio y proporcionar los datos necesarios para su posterior procesado. El conjunto de algoritmos y transformaciones necesarias para realizar el análisis y obtener la información de la imagen es el software. Dicha información puede ser utilizada para su representación visual, para actuar en el controlador de un robot o pueden ser los datos para un autómata programable. En definitiva, múltiples periféricos pueden ser receptores de esta información y vincularse con el sistema de procesamiento de las imágenes.

Las peculiaridades de cada subsistema se pueden resumir de la siguiente manera:

**SUBSISTEMA DE ILUMINACIÓN:** conjunto de artefactos que producen radiación electromagnética que incide sobre los objetos a visualizar. Por ejemplo: lámparas, pantallas fotográficas, filtros de luz, láseres...

**SUBSISTEMA DE CAPTACIÓN:** son los transductores que convierten la radiación luminosa reflejada en señales eléctricas. Fundamentalmente se habla de sensores CCD o CMOS, no sólo en el espectro visible, sino que van desde la radiación gamma hasta la radiofrecuencia o microondas, dando paso a sensores de ultrasonidos, sonar, radar...

**SUBSISTEMA DE ADQUISICIÓN:** la señal eléctrica procedente las cámaras forma la señal de video. Hay una tendencia creciente a que su naturaleza sea de tipo digital, pero todavía existen muchas señales de carácter analógico. Para ser tratadas, hay que

muestrearlas y cuantificarlas. Ambas tareas son realizadas por las tarjetas de adquisición; también llamadas "frame grabbers". Existen diferentes tipos de buses desde PCI hasta VMP. Recientemente, también se están empleando las tecnologías USB o FireWire.

**SUBSISTEMA DE PROCESAMIENTO:** suele ser un ordenador o un clúster de ordenadores, dependiendo de las necesidades de los algoritmos de visión artificial. Parten de una representación digital de las imágenes y procesan esta información hasta alcanzar otro tipo de información de más alto nivel. La transformación dependerá de la algoritmia utilizada.

**SUBSISTEMAS PERIFÉRICOS:** conjunto de elementos receptores de la información de alto nivel. Puede ser un monitor de altas prestaciones gráficas, un automatismo, una impresora mostrando las características...

- ***Etapas de un proceso de visión computacional***

Las etapas de un proceso de visión artificial se suelen dividir en dos grandes grupos. Aquellas que ejecutan métodos de bajo nivel, encargadas de obtener las características básicas de la imagen (bordes, regiones...) y las que realizan un procesamiento de alto nivel, que basándose en las características básicas del nivel anterior describen la escena.

A continuación se muestra un esquema de las etapas involucradas en este proceso.

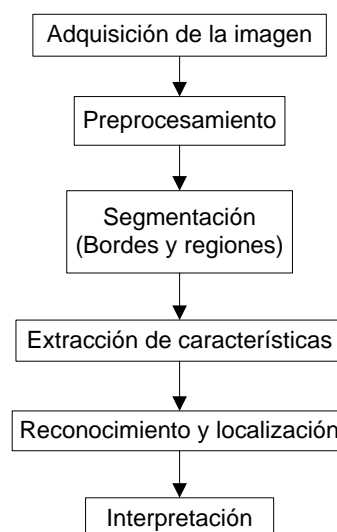


Figura 3 Etapas de un proceso de visión computacional

## 2. APROXIMACIÓN AL PROBLEMA

### 2.1 Sistema de visión dinámica

Una escena dinámica se compone de una secuencia de fotogramas interrelacionados. Cada fotograma, representa la escena en un determinado instante de tiempo.

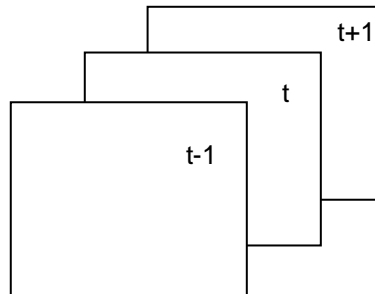


Figura 4 Secuencia de fotogramas

#### ▪ **Subsistemas**

Un sistema de visión dinámica está compuesto por cuatro subsistemas: procesamiento de imagen, modelo cinemático, identificación y predicción de trayectoria y control de ejes.

El subsistema de procesamiento de imagen analiza la secuencia que proporciona la cámara, segmentando las zonas que corresponden al fondo de la escena de las que corresponden al objeto en movimiento. La información generada es básicamente la posición del móvil.

El modelo cinemático del conjunto cámara-óptica recibe las coordenadas de la posición del móvil, calculadas previamente por el sistema de visión.

La información que generan los módulos anteriores sobre la trayectoria del móvil no es válida para ser empleada directamente como parámetro de control del sistema debido a los siguientes factores:

- Ruido de medida producido por los errores de los sensores (cámara y encoders).
- Errores de modelización dinámica y cinemática.
- Retardo respecto a la trayectoria real del móvil; provocado, fundamentalmente, por el tiempo de respuesta del sistema de tratamiento de imágenes.



Con la finalidad de evitar los efectos negativos de los factores anteriores, el módulo de identificación y predicción de la trayectoria deberá, necesariamente, realizar las siguientes funciones:

- Filtrar el ruido de medida.
- Compensar los errores de modelización.
- Predecir la trayectoria del móvil para compensar los retardos producidos por el sistema de visión.

Finalmente, el sistema de control calcula las consignas de movimiento de los ejes para situarlos en la posición establecida por el sistema de identificación de la trayectoria.

▪ **Variables en un sistema de visión dinámica**

En general, la entrada de un sistema de visión dinámica es una secuencia de imágenes que puede expresarse como una función de tres o cuatro variables,  $F(x, y, t)$  o  $F(x, y, c, t)$ , que se resumen en la tabla siguiente.

VARIABLES	DESCRIPCIÓN
$x, y$	Coordenadas espaciales en el fotograma
$c$	Canal de color
$t$	Instante de tiempo

Figura 5 Tabla de variables de un sistema de visión dinámica

## 2.2 Análisis de escenas dinámicas

Los cambios en la escena puede deberse al movimiento en la propia escena, al movimiento de la cámara o a ambos. En cualquier caso, el análisis de escenas dinámicas, al igual que las estáticas, se divide en tres fases:

- Análisis periférico: extracción de información aproximada
- Análisis de atención: importancia en las partes activas de la escena
- Análisis cognitivo: aplicación del conocimiento disponible para examinar la secuencia

Los pasos básicos son la segmentación y el análisis, que son esencialmente algoritmos de software y funciones que se ejecutan en el procesador.

### 3. VISIÓN DINÁMICA

#### 3.1 Métodos de detección del movimiento

Siempre que trabajamos con vídeo (tanto en tiempo real como pre-grabado) y queremos saber qué sucede en la secuencia (movimiento de la cámara o de los objetos), parte del problema se basa en saber hacia dónde se mueven ciertos puntos característicos en un fotograma con respecto a los siguientes o anteriores.

Una posible clasificación de los métodos de estimación, se realiza de acuerdo a si llevan a cabo una estimación del movimiento del plano de la imagen (movimiento 2D) o bien una estimación del movimiento de objetos (movimiento 3D, donde el tiempo es la tercera dimensión).

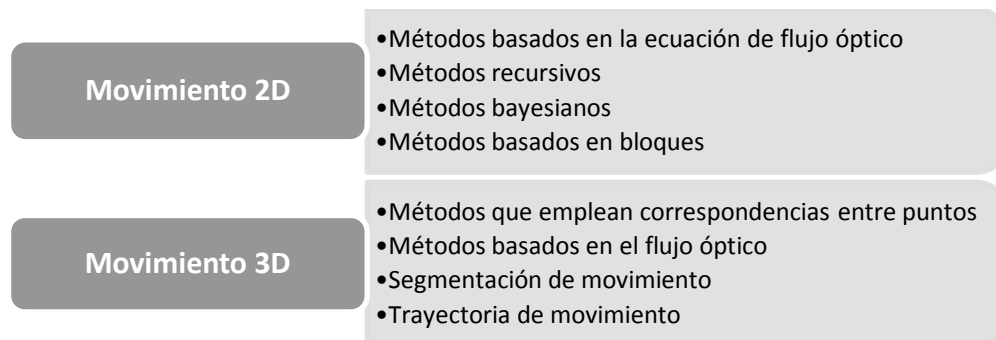


Figura 6 Clasificación de los métodos de estimación de movimiento

La mayoría de las técnicas de análisis de escenas dinámicas están basadas en la detección de cambios en una secuencia de imágenes, ya sea entre imágenes, regiones o píxeles. Un movimiento perceptible en la escena provocará cambios en la secuencia de imágenes, por lo que las características del movimiento podrán analizarse si dichos cambios son detectados. Por el contrario, cualquier cambio en la iluminación de la escena, aunque puede provocar cambios en los niveles de intensidad de la imagen, no se convertirá en un movimiento real de los objetos.

Algunas de las técnicas de detección de movimiento más usuales son:

- Sustracción del fondo
- Imagen diferenciada
- Imagen de diferencias acumuladas
- Ajuste de bloques
- Detección de cambios en el flujo óptico

### 3.2 Detección de cambios

- **Sustracción de fondo**

La sustracción de fondo se realiza comparando las imágenes de las que está compuesto un video. Dada una imagen inicial, para el instante  $t=0$ , y una vez realizadas las conversiones pertinentes, se van restando una por una las imágenes posteriores y de este modo, se puede comprobar si se ha producido algún cambio (movimiento). La función a implementar en Matlab es la siguiente:

$$Z = imabsdiff(X,Y)$$

Donde Z es la imagen resultante de la diferencia absoluta ("imabsdiff") de dos imágenes X e Y.

Para evitar el ruido y facilitar la visualización, la secuencia de imágenes se umbraliza y se convierte a binaria. Las funciones de Matlab que realizan estas operaciones tienen la siguiente sintaxis:

$$level = graythresh(I)$$

$$J = im2bw(I, level)$$

Donde "level" es el umbral, "I" es la imagen original y "J" la imagen umbralizada. La función "graythresh" devuelve el umbral óptimo de la imagen utilizando el método de Otsu. Hay que destacar que el umbral es devuelto en el rango [0,1] por lo que hay que binarizar la imagen. La función "im2bw" convierte a negro todo lo que sea mayor que el umbral y a blanco todo lo que sea menor.

Esta técnica se emplea cuando hay una clara diferencia entre los objetos a extraer respecto del fondo de la escena. Los principios que rigen son la similitud entre los píxeles pertenecientes a un objeto y sus diferencias respecto al resto. Por tanto, la escena debe caracterizarse por un fondo uniforme y por objetos parecidos.

- **Imagen diferencia y diferencias acumuladas**

La simple comparación entre dos imágenes resulta un método obvio de detección de cambios. Esto se implementa como una resta en valor absoluto que se puede umbralizar para obtener así una imagen binaria. Se usa fundamentalmente con cámara fija o fondo estático aunque presenta diversos problemas, como que los movimientos lentos pueden perderse.

La imagen de diferencias acumuladas (ADP) se forma comparando una secuencia de "frames" con la imagen de referencia, incrementando el valor en la ADP si se sobrepasa un umbral para la diferencia de intensidad de los píxeles. Al comparar mayor número de "frames" se consigue información más fiable evitando ruido casual y, por lo tanto, solventando los problemas con los objetos pequeños o movimientos lentos.

- **Ajuste de bloques o block matching**

Otra técnica para la detección del movimiento es el ajuste de bloques. Como su nombre indica, se divide la imagen en bloques y se estima el desplazamiento bajo una medida de error. El desplazamiento que minimice el error se considera el movimiento de la imagen.

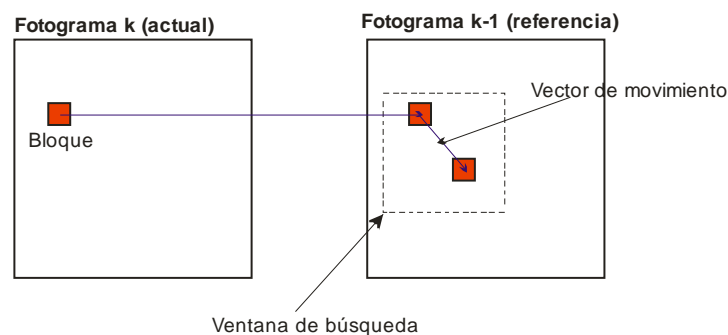


Figura 7 Esquema básico del algoritmo block-matching

Como puede verse en la figura, los bloques del fotograma actual se comparan con los bloques del fotograma de destino o de referencia (anterior al actual, generalmente el primero), deslizando el actual a lo largo de una región concreta de píxeles del fotograma de destino. Un criterio de semejanza determina la elección del bloque con mayor similitud (o que minimiza un error medido) de entre los candidatos dentro de la ventana de búsqueda de tamaño fijo del fotograma de referencia. Si el bloque elegido no se encuentra en la misma posición en ambos frames, significa que se ha movido. La distancia del bloque coincidente entre el fotograma actual y el de referencia se define como el vector de desplazamiento estimado, y será el que se le asigne a todos los píxeles del macrobloque.

Los requerimientos deseables para un buen algoritmo de "block matching" o ajuste de bloques son la convergencia, pocos puntos de búsqueda, pocas iteraciones e inmunidad respecto al ruido. Además hay otros criterios importantes como la complejidad del hardware.

Una de las técnicas utilizadas es el método "Three-step". Su nombre se debe a que la estimación de movimientos se realiza en tres iteraciones.

Evidentemente, ningún algoritmo podrá cumplir todos los requerimientos simultáneamente pero el método "conjunto modificado" ofrece unas buenas prestaciones. El número de iteraciones necesarias para estimar el desplazamiento varía entre 4 y 8, mientras que el número de puntos examinados está comprendido entre 9 y 13. En simulaciones prácticas, se ha obtenido que éste método es sensiblemente más rápido que otros de ajuste de bloques y tarda casi la mitad que el algoritmo descrito anteriormente.

### 3.3 Segmentación

Se trata de decidir qué partes de la imagen necesitan interpretación o análisis y cuáles no, es decir, consistirá en separar las componentes dinámicas de las estáticas. Puede ser difícil en casos donde la cámara esté en movimiento puesto que extraer componentes estáticas dependerá del conocimiento del movimiento del sistema de referencia.

A menudo es posible y necesario refinar la segmentación. Por ejemplo, si la aplicación debe encontrar arañazos o grietas en un vidrio, la primera fase de segmentación suele encontrar elementos que son las propias grietas o arañazos además de suciedad y polvo. Frecuentemente se perfecciona este resultado para eliminar de la imagen la suciedad y el polvo de los elementos a analizar.

Las técnicas más usuales de segmentación en secuencias de imágenes son detección de bordes y regiones.

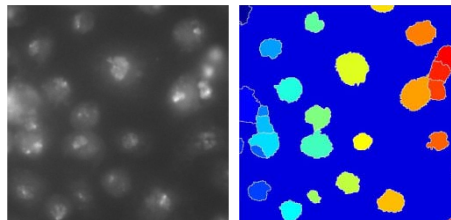


Figura 8 Ejemplo segmentación (<http://blogs.mathworks.com/steve/2006/06/02/cell-segmentation/>)

En la detección de bordes, la idea es combinar gradientes espaciales y temporales usando un operador lógico AND. Aplicando un valor umbral al producto en lugar de a cada uno de los factores se detectarán los bordes en movimiento de manera más robusta. Otro método para detectar bordes es usar la función "edge" sobre las imágenes en escala de grises y escogiendo el tipo de filtro adecuado a las imágenes en cuestión (Sobel, Prewitt, Roberts, Canny...). También es posible segmentar una escena con poca carga computacional usando las imágenes de diferencias acumuladas.

### 3.4 Correspondencia

El ajuste o correspondencia trata de hacer corresponder de forma coherente un punto  $p_i$  de una imagen con otro  $p_j$  de la imagen siguiente. Para ello, es necesario plantearse dos cuestiones:

- Cuáles son las características a elegir, suelen ser puntos representativos (bordes, esquinas...)
- Cuáles son las restricciones a imponer, por ejemplo la distancia máxima entre conexiones

Las propiedades que caracterizan el ajuste son:

- Discretización: distinción de puntos individuales
- Similitud: semejanza de dos puntos emparejados, distancias de emparejado
- Consistencia: calidad del ajuste

### 3.5 Flujo óptico

Se denomina campo de movimiento (motion field) al campo tridimensional de velocidad del movimiento relativo a los objetos y el observador en un espacio. Es decir, la información de dirección y magnitud de movimiento en todos los puntos de una región determinada. La porción visible de la proyección del campo de movimiento sobre un plano imagen se denomina flujo de imagen (image flow). Las técnicas de flujo óptico (optical flow) estiman el flujo de imagen operando sobre los cambios de intensidad en una secuencia de imágenes.

La estimación del movimiento a través del cambio de intensidades presenta defectos intrínsecos (tanto en sistemas de visión biológica como artificial), que en determinados casos provocarán que la estimación se aparte de los vectores de movimiento reales de los objetos. En sistemas de visión biológica, estas limitaciones dan lugar a ciertos fenómenos de ilusión óptica.

Algunas aplicaciones del flujo óptico son: navegación autónoma, compresión y descompresión de video y estudio dinámico de objetos.

- **Problemas: iluminación y apertura**

El flujo óptico varía en función de cómo se ilumine el objeto a estudiar. La apertura es un problema inherente a visión, tanto biológica como artificial. Surge de la extracción insuficiente de información en el entorno.

- ***Ecuación de restricción del flujo óptico (ERFO)***

Se establecen dos supuestos iniciales:

- Los cambios de intensidad se deben exclusivamente al movimiento.
- Un objeto cualquiera mantendrá su nivel de intensidad después de un desplazamiento.

Matemáticamente se puede demostrar que solo se puede extraer la componente normal de movimiento de los bordes espaciales de intensidad constante.

- ***Técnicas basadas en gradiente***

Las técnicas basadas en gradiente tratan de calcular la velocidad a partir de las derivadas espacio temporales de la intensidad de una imagen. Se basan por tanto en suposiciones sobre la forma en que cambia la intensidad de la imagen según las coordenadas y con respecto al tiempo. Tiene la ventaja de que genera un campo vectorial muy denso (idealmente un vector por píxel) que resulta muy útil en algunas aplicaciones. Sin embargo, adolece del problema de que es muy sensible al ruido en las imágenes y que no puede determinar el flujo bajo ciertas circunstancias.

Los algoritmos más empleados son los siguientes:

- Horn & Schunck
- Lucas-Kanade
- Nagel

- ***Técnicas basadas en correlación***

Se destacan dos algoritmos basados en técnicas de correlación:

- Anandan
- Ajit Singh

- ***Técnicas basadas en frecuencia***

En vez de comparar la información de brillo en los píxeles como hace el "block matching", se realiza un análisis en frecuencia del bloque. El uso de la transformada de Fourier es vital en el concepto de la correlación de fase. Una línea de luminancia, que en el dominio digital está compuesta de una serie de muestras, es una función del brillo que hay en la imagen. La transformada de Fourier convierte esta función en un espectro de amplitudes de frecuencia y fases. En el dominio espacial, una diferencia de fase se corresponde con un movimiento físico.

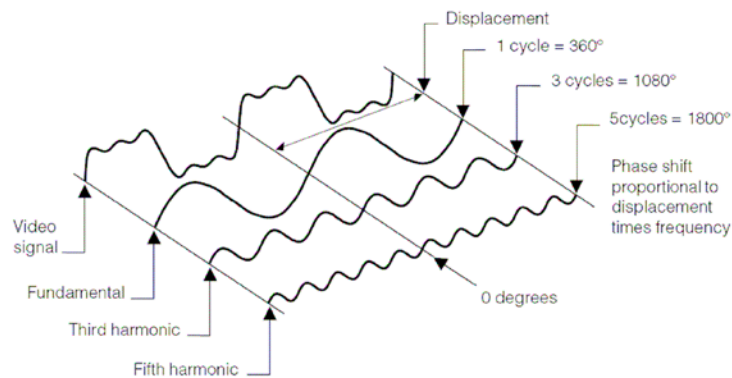


Figura 9 Desplazamiento en la fase en una señal de vídeo

Los siguientes algoritmos usan el dominio de la frecuencia para estimar el movimiento relativo entre dos imágenes:

- Fleet & Jepson
- Heeger

### 3.6 Seguimiento

El seguimiento de un único objeto es sencillo, mas la cosa se complica si hay que analizar varios móviles (problema de correspondencia). Su objetivo principal es el seguimiento espacio-temporal de un objeto a través de la estimación óptima de su estado. Se trata de predecir y actualizar el estado futuro del móvil, estas operaciones son típicas de filtros. Entre otros, se destaca el filtro de Kalman y el de partículas.

#### ▪ **Coherencia del camino**

A la hora de resolver el problema de correspondencia o coherencia del camino en una secuencia de imágenes, se asume lo siguiente:

- La posición de un punto dado será relativamente invariante de un "frame" al siguiente.
- La velocidad escalar de un punto dado será relativamente invariante de un "frame" al siguiente.
- La dirección de movimiento de un punto dado relativamente invariante de un "frame" al siguiente.

#### ▪ **Filtro Kalman**

El Filtro de Kalman es una técnica recursiva para determinar los parámetros correctos de un sistema que evoluciona con el tiempo. Fue inventado por Rudolph Emil



Kalman a finales de la década de 1950, con la finalidad de filtrar y predecir sistemas lineales.

Básicamente es un conjunto de ecuaciones matemáticas que implementan un estimador óptimo del tipo predictor-corrector.

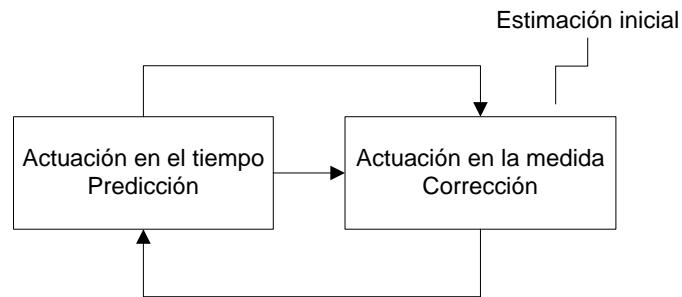


Figura 10 Filtro Kalman

Para aplicar el filtro de Kalman al seguimiento de un objeto, es necesario proporcionar una característica representativa del mismo, que será tomada como la observación del objeto. Para el cálculo de este punto que denominaremos centro de masas y que determina la posición del objeto, es necesario realizar una serie de operaciones sobre la imagen: detección de movimiento, segmentación y determinación de la posición

Para estimar el valor actual de las variables de interés son necesarios el conocimiento del sistema y dispositivos dinámicos de medida, la descripción estadística de los ruidos del sistema, errores de medida e incertidumbre en los modelos dinámicos, y cualquier información disponible acerca de las variables de interés.

Como ya se ha dicho, este filtro es un algoritmo de procesamiento de datos, y no es más que un programa de ordenador. Por ello, tendrá que trabajar sobre medidas discretas en vez de trabajar en tiempo continuo. Además, al ser un algoritmo recursivo, no requiere ir almacenando todos los datos previos para procesarlos de nuevo cada vez que se tome una nueva medida.

- **Filtro de partículas**

Los filtros de partículas son estimadores secuenciales de Monte Carlo aplicables a cualquier transición de estados o modelos de medida. El propósito general de estos algoritmos es representar funciones de densidad de probabilidad y su evolución en el tiempo. Estas representaciones están basadas en muestras discretas de las funciones que son modeladas, las partículas, que están formadas por un estado ( $x$ ) y un peso ( $\pi$ ).

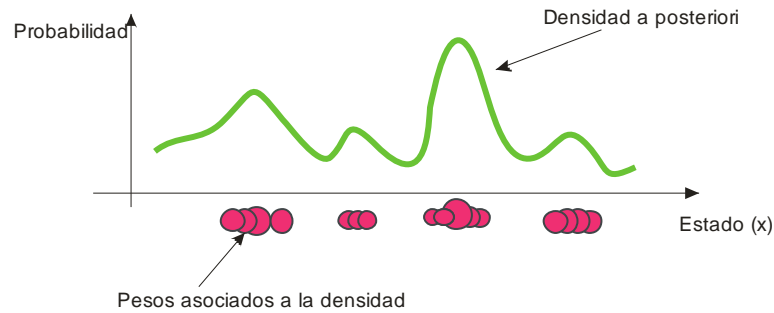


Figura 11 Ejemplo de muestreo y pesos obtenidos al aplicar el filtro de partículas

La evolución temporal de las partículas se lleva a cabo mediante un procedimiento iterativo de actualización – predicción.

Un filtro de partículas consta de 5 partes: inicialización, evaluación, selección, difusión y predicción del movimiento. La inicialización sólo se ejecuta 1 vez, el resto se ejecutan en un bucle. A continuación se explica cada una de las partes:

1. Inicialización: consiste muestrear un número inicial de partículas con posiciones aleatorias sobre una imagen dando como peso a cada una de dichas partículas el valor 0.
2. Evaluación: consiste en calcular el peso de cada partícula para posteriormente hacer un estudio de todas las partículas con sus pesos y descubrir en que zona están las partículas que nos interesan.
3. Selección: consiste en obtener un histograma con la acumulación de los pesos de las partículas. Luego sobre este histograma lanzaremos una serie de pesos aleatorios del histograma para comprobar a que partícula pertenece, y así obtener las partículas que más nos interesan.
4. Difusión: como cada partícula puede ser elegida más de una vez, se aplica un desplazamiento aleatorio a cada una para evitar coincidencias.
5. Movimiento: aplica el modelo del sistema para construir el conjunto de partículas en el instante siguiente.

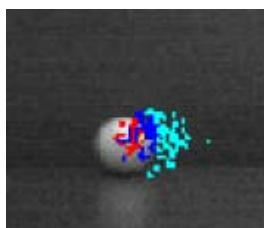


Figura 12 Resultado de una prueba realizada con una pelota

## 4. HERRAMIENTAS Y LIBRERÍAS SOFTWARE

En este apartado se muestra algún software actual disponible y que se utiliza para el desarrollo de las diversas implementaciones de video tracking.

### 4.1 Plataformas de desarrollo de contenidos

Software enfocado a la edición y creación de contenidos 3D. Es decir, aquí se exponen algunos editores que mediante una interfaz de usuario permiten modelar objetos, aplicar texturas, definir la iluminación de una escena, realizar diferentes tipos de rendering... etc. Están basados en tecnologías de graphics pipeline como OpenGL y Direct 3D.

- Autodesk Maya  
[www.autodesk.com/maya](http://www.autodesk.com/maya)
- 3D Studio MAX  
[www.autodesk.com/3dsmax](http://www.autodesk.com/3dsmax)
- Blender (software libre multiplataforma)  
[www.blender.org](http://www.blender.org)



Figura 13 Interfaz de Blender

### 4.2 Software de integración y tracking de cámara

Algunas plataformas de desarrollo de contenidos incluyen herramientas que facilitan la composición de imágenes sintéticas con escenas de vídeo real previamente rodadas. Pero también existe un gran mercado de añadidos software (o plug-ins) que ofrecen mayor precisión y soluciones mejoradas enfocadas al ámbito profesional.

Entre otros:

- Boujou  
[www.2d3.com](http://www.2d3.com)
- 3D Equalizer  
[www.3dequalizer.com](http://www.3dequalizer.com)
- Voodoo Tracker  
<http://www.digilab.uni-hannover.de/download.html>
- MatchMover Pro  
<http://sfx.realviz.com/products/mpro/index.php>

### **4.3 Librerías de visión artificial y tracking de puntos**

Son las librerías de código abierto utilizadas actualmente en los campos de la realidad aumentada y visión artificial. Sirven como fuente de algoritmos codificados en diversos lenguajes de programación.

Por ejemplo:

- ARToolkit  
<http://www.hitl.washington.edu/artoolkit/>
- KLT Tracker  
<http://www.ces.clemson.edu/~stb/klt/>

## 5. APLICACIONES

El número de aplicaciones relacionadas con la visión artificial aumenta cada día. A continuación se citan algunos de los campos donde es empleada esta disciplina, y en particular, el seguimiento de imágenes:

- Captura de movimiento
- Interacción hombre-máquina
- Vigilancia
- Reconocimiento a partir del movimiento

El “video tracking” no es solamente útil para monitorizar la actividad en lugares públicos sino que se está convirtiendo en una parte clave para el posterior análisis de imágenes de video. Por ejemplo, la información sobre la situación e identidad de objetos en diferentes instantes de tiempo es la base para detectar posibles movimientos inusuales de éstos – si alguien es asaltado en un cajero automático– o para coordinar actividades –como jugadas estratégicas en un partido de fútbol–.

CAMPO DE APLICACIÓN	PROCESOS/OPERACIONES
<b>APLICACIONES INDUSTRIALES</b> Automoción Electrónica Alimentación Farmacia ...	Control de calidad Posicionado de piezas Inspección y detección de fallos/defectos Metrología industrial sin contacto Trazabilidad Guiado de robots Control de soldaduras Identificación de piezas/componentes Control de envasado de blisters, frascos, cápsulas...
<b>VIDEO-VIGILANCIA</b>	Tráfico viario Matrículas de coches Seguimiento de personas Reconocimiento de objetos/caras
<b>MEDICINA</b>	Análisis de imágenes de microscopía
<b>DEPORTES</b>	Análisis de técnicas deportivas
<b>ASTRONOMÍA</b>	Exploración del espacio
<b>METEOROLOGÍA</b>	Predicción del tiempo
<b>OCIO</b>	Películas Video juegos

Figura 14 Campos de aplicación de la visión dinámica

## 6. CONCLUSIONES

En la presente memoria, dada su corta extensión, se ha tratado de ofrecer una visión general de los diferentes métodos y tecnologías que pueden aplicarse al campo de la visión artificial en la tarea de seguimiento de objetos. En primer lugar, se definen los subsistemas y variables presentes en un sistema de visión dinámica.

Para poder realizar con éxito el seguimiento de objetos es preciso conocer previamente otras técnicas útiles a la hora de detectar el movimiento, como son la sustracción del fondo o el block-matching -a la hora de detectar cambios-, o técnicas de flujo óptico.

Cabe destacar la importancia de seleccionar una buena técnica de segmentación de los objetos de interés, ya que esta tarea es crucial a la hora de realizar un buen seguimiento de los mismos. Si el objeto no se logra segmentar, difícilmente podrá seguirse.

Particularizando a seguimiento de imágenes el video se describen dos algoritmos diferentes: el filtro Kalman y el filtro de partículas.

El filtro de partículas representa una poderosa herramienta para el tratamiento de procesos del mundo real, evitando hacer cualquier tipo de suposiciones sobre las características intrínsecas del proceso como hacen los filtros clásicos (Kalman). Su funcionamiento permite centrar la atención en una región de la imagen donde es posible encontrar al objeto, despreciando la exploración del resto. La diferencia fundamental entre ambos algoritmos radica en el hecho de que los filtros Kalman pueden representar solamente la estimación del estado por una distribución gaussiana unimodal, mientras que los filtros de partículas pueden representar densidades multimodales complejas empleando una gran cantidad de partículas aleatoriamente muestreadas.

Otro estudio de interés sería desarrollar el proceso de seguimiento con el uso de cámaras móviles, como por ejemplo las utilizadas en dispositivos de seguridad o vigilancia, donde los movimientos propios de la cámara pueden estar restringidos y controlados.

## 7. RECURSOS DE INTERÉS

### ▪ **Artículos**

Fernando Aguado, Begoña; Alferes Castaño, Janelcy; Zalama Casanova, Eduardo; Gómez García-Bermejo, Jaime. Diseño y simulación de un filtro Kalman para un robot móvil. 2004

### ▪ **Videos**

- Seguimiento de manos:

<http://www.youtube.com/watch?v=ln74UoHeiPg>

- Seguimiento facial en tiempo real:

<http://www.youtube.com/watch?v=P5uxMIhItMg>

<http://svr-www.eng.cam.ac.uk/research/vision/research/vistrack.html>

- Aplicación del filtro Kalman a una avioneta:

<http://www.youtube.com/watch?v=0GSIKwfkFCA&feature=related>

- OpenCV Kalman Filter:

<http://www.univision.com/uv/video/Opencv-Kalman-Filter/id/2751058141>

### ▪ **Códigos**

- Sintetizador de Vídeo [VideoSyn](#)

Se incluye el código fuente incluido en un proyecto de Eclipse además de una carpeta con ficheros necesarios para las pruebas. Para compilar el proyecto son necesarias las librerías opencv y cppunit.

- Filtro de partículas básico [BasePF](#)

Se incluye el código fuente incluido en un proyecto de Eclipse además de una carpeta con ficheros necesarios para las pruebas. Para compilar el proyecto son necesarias las librerías opencv y cppunit.

- G. Welch, G. Bishop. An Introduction to the Kalman Filter. TR 95-041. UNC-Chapel Hill, <http://www.cs.unc.edu/~welch/kalman/>

Tutorial en español sobre el filtro de Kalman

### ▪ **Noticias relacionadas**

<http://www.larazon.es/noticia/vision-artificial-en-tiempo-real-contra-la-baja-vision>

## 8. BIBLIOGRAFÍA

Platero Dueñas, Carlos. **Apuntes de visión artificial**. Universidad Politécnica de Madrid, 2008

[http://en.wikipedia.org/wiki/Video\\_tracking](http://en.wikipedia.org/wiki/Video_tracking)

<http://www.matpic.com/>

<http://www.cs.unc.edu/~welch/kalman/>

<http://www.cmucam.org/>

<http://gavab.escet.urjc.es/master/visionartificial>

<http://homepages.inf.ed.ac.uk/rbf/CVonline/>

[http://www.qubitsystems.com/Merchant2/merchant.mvc?Screen=CTGY&Store\\_Code=QS&Category\\_Code=APB-05](http://www.qubitsystems.com/Merchant2/merchant.mvc?Screen=CTGY&Store_Code=QS&Category_Code=APB-05)

[http://www.harvardapparatus.com/webapp/wcs/stores/servlet/haicat2\\_10001\\_11051\\_41301\\_-1\\_HAI\\_Categories\\_N\\_37367](http://www.harvardapparatus.com/webapp/wcs/stores/servlet/haicat2_10001_11051_41301_-1_HAI_Categories_N_37367)

<http://www.tigoe.net/pcomp/videoTrack.shtml>

<http://www.ge-ip.com/products/family/video-tracking-and-image-processing>

<http://www.mediaeater.com/cameras/face.html>